aws

Guida per l'utente

CloudWatch Registri Amazon

CloudWatch Registri Amazon: Guida per l'utente

Table of Contents

Che cos'è Amazon CloudWatch Logs?

Puoi utilizzare Amazon CloudWatch Logs per monitorare, archiviare e accedere ai tuoi file di log da AWS CloudTrailistanze Amazon Elastic Compute Cloud (Amazon EC2), Route 53 e altre fonti.

CloudWatch Logs ti consente di centralizzare i log di tutti i sistemi, le applicazioni e i AWS servizi che utilizzi, in un unico servizio altamente scalabile. Potrai quindi visualizzarli facilmente, cercarli per codici o modelli di errore specifici, filtrarli in base a campi specifici o archiviarli in modo sicuro per analisi future. CloudWatch I registri consentono di visualizzare tutti i registri, indipendentemente dalla loro origine, come un flusso unico e coerente di eventi ordinati per ora.

CloudWatch Logs supporta anche l'interrogazione dei log con un potente linguaggio di query, il controllo e il mascheramento dei dati sensibili nei log e la generazione di metriche a partire dai log utilizzando filtri o un formato di registro incorporato.

Funzionalità

- Interroga i dati di registro: puoi utilizzare CloudWatch Logs Insights per cercare e analizzare in modo interattivo i tuoi dati di registro. È possibile eseguire interrogazioni per rispondere in modo più efficiente ed efficace ai problemi operativi. CloudWatch Logs Insights include un linguaggio di interrogazione creato appositamente con pochi comandi semplici ma potenti. Per iniziare, forniamo query di esempio, descrizioni dei comandi, completamento automatico delle query e individuazione dei campi di log. Sono incluse query di esempio per diversi tipi di registri di servizio. AWS Per iniziare, consulta Analisi dei dati di registro con CloudWatch Logs Insights.
- Rileva ed esegui il debug con Live Tail: puoi usare Live Tail per risolvere rapidamente gli incidenti visualizzando un elenco di streaming dei nuovi log eventi man mano che vengono importati. Puoi visualizzare, filtrare ed evidenziare i log importati quasi in tempo reale, in modo da poter rilevare e risolvere rapidamente i problemi. Puoi filtrare i log in base ai termini specificati e anche evidenziare quelli che contengono termini specifici per in modo da poter trovare rapidamente ciò che stai cercando. Per ulteriori informazioni, consulta Utilizzo di Live Tail per visualizzare i log quasi in tempo reale.
- Monitora i log dalle istanze Amazon EC2: puoi CloudWatch utilizzare Logs per monitorare applicazioni e sistemi utilizzando i dati di registro. Ad esempio, CloudWatch Logs può tenere traccia del numero di errori che si verificano nei log delle applicazioni e inviarti una notifica ogni volta che il tasso di errori supera una soglia specificata. CloudWatch Logs utilizza i dati di registro per il monitoraggio, quindi non sono necessarie modifiche al codice. Ad

esempio, è possibile monitorare i registri delle applicazioni per termini letterali specifici (come "NullReferenceException«) o contare il numero di occorrenze di un termine letterale in una posizione particolare nei dati di registro (come i codici di stato «404" in un log di accesso Apache). Quando viene trovato il termine che state cercando, CloudWatch Logs riporta i dati in base a una metrica specificata dall'utente. CloudWatch I dati di log vengono crittografati durante il transito e mentre sono a riposo. Per iniziare, consulta Guida introduttiva ai CloudWatch registri.

- Monitora gli eventi AWS CloudTrail registrati: puoi creare allarmi CloudWatch e ricevere notifiche relative a particolari attività API acquisite CloudTrail e utilizzare la notifica per eseguire la risoluzione dei problemi. Per iniziare, consulta Invio di CloudTrail eventi ai CloudWatch registri nella Guida per l'AWS CloudTrail utente.
- Verifica e maschera i dati sensibili: se hai dati sensibili nei tuoi log, puoi contribuire a salvaguardarli con policy di protezione dei dati. Queste policy consentono di controllare e mascherare i dati di log sensibili. Se abiliti la protezione dei dati, i dati sensibili che corrispondono agli identificatori dei dati che hai selezionato vengono mascherati per impostazione predefinita. Per ulteriori informazioni, consulta Incremento della protezione dei dati di log sensibili con il mascheramento.
- Conservazione dei log: per impostazione predefinita, i log vengono conservati a tempo indeterminato e non scadono mai. Puoi modificare la policy di conservazione per ogni gruppo di log mantenendo la conservazione a tempo indeterminato o scegliendo periodi di conservazione compresi tra 10 anni e un giorno.
- Archivia i dati di registro: puoi utilizzare CloudWatch Logs per archiviare i dati di registro in uno spazio di archiviazione altamente durevole. L'agente CloudWatch Logs semplifica l'invio rapido di dati di registro ruotati e non ruotati da un host al servizio di registro. Puoi quindi accedere ai dati di log grezzi in caso di necessità.
- Registra le interrogazioni DNS di Route 53: puoi utilizzare CloudWatch Logs per registrare le informazioni sulle query DNS ricevute da Route 53. Per ulteriori informazioni, consulta Registrazione delle query DNS nella Guida per sviluppatori di Amazon Route 53.

Servizi correlati AWS

I seguenti servizi vengono utilizzati insieme ai registri: CloudWatch

- AWS CloudTrailè un servizio web che consente di monitorare le chiamate effettuate all'API CloudWatch Logs per l'account, incluse le chiamate effettuate da AWS Management Console, AWS Command Line Interface (AWS CLI) e altri servizi. Quando CloudTrail la registrazione è attivata, CloudTrail acquisisce le chiamate API nel tuo account e invia i file di registro al bucket

Amazon S3 specificato. Ogni file di log può contenere uno o più record, in base al numero di operazioni da eseguire per soddisfare una richiesta. Per ulteriori informazioni su, consulta What AWS CloudTrailIs? AWS CloudTrail nella GuidaAWS CloudTrail per l'utente. Per un esempio del tipo di dati che vengono CloudWatch scritti nei file di CloudTrail registro, vedereRegistrazione delle chiamate API del File di log Amazon CloudWatch in AWS CloudTrail.

- AWS Identity and Access Management (IAM) è un servizio web che ti aiuta a controllare in modo sicuro l'accesso alle AWS risorse per i tuoi utenti. Utilizza IAM per controllare chi può utilizzare le tue risorse AWS (autenticazione), quali risorse e in che modo (autorizzazione). Per ulteriori informazioni, consulta Che cos'è IAM? nella Guida per l'utente di IAM.
- Amazon Kinesis Data Streams è un servizio Web che puoi utilizzare per l'acquisizione e l'aggregazione di dati rapidamente e continuamente. Il tipo di dati utilizzato include dati di logo dell'infrastruttura IT, log di applicazioni, social media, feed di dati di mercato e dati clickstream Web. Poiché il tempo di risposta per il consumo e l'elaborazione dei dati è in tempo reale, l'elaborazione è in genere leggera. Per ulteriori informazioni, consulta Cos'è Amazon Kinesis Data Streams? nella Guida per gli sviluppatori di Amazon Kinesis Data Streams.
- AWS Lambda è un servizio Web che puoi utilizzare per creare applicazioni che rispondono rapidamente a nuove informazioni. Carica il codice dell'applicazione come funzioni Lambda e Lambda eseguirà il codice in un'infrastruttura di calcolo ad alta disponibilità ed eseguirà tutte le attività di amministrazione delle risorse di calcolo, tra cui la manutenzione del server e del sistema operativo, il provisioning della capacità e la scalabilità automatica, la distribuzione di patch di sicurezza e per il codice e il monitoraggio e la registrazione del codice. Tutto quello che occorre fare è fornire il proprio codice in una delle lingue supportate da Lambda. Per ulteriori informazioni, consulta Cos'è AWS Lambda? nella Guida per gliAWS Lambda sviluppatori.

Prezzi

Quando ti registri AWS, puoi iniziare a usare CloudWatch Logs gratuitamente utilizzando il pianoAWS gratuito.

Le tariffe standard si applicano ai log archiviati da altri servizi che utilizzano CloudWatch Logs (ad esempio, i log di flusso di Amazon VPC e i log Lambda).

Per ulteriori informazioni sui prezzi, consulta la pagina CloudWatch dei prezzi di Amazon.

Per ulteriori informazioni su come analizzare i costi e l'utilizzo di CloudWatch Logs e CloudWatch per le best practice su come ridurre i costi, consulta CloudWatch Fatturazione e costi.

Concetti di Amazon CloudWatch Logs

La terminologia e i concetti fondamentali per la comprensione e l'uso di CloudWatch Logs sono descritti di seguito.

Eventi di log

Un evento di log è un record di alcune attività registrate dall'applicazione o dalla risorsa monitorata. Il record degli eventi di registro che CloudWatch Logs comprende contiene due proprietà: il timestamp di quando si è verificato l'evento e il messaggio di evento non elaborato. I messaggi di evento devono essere in codifica UTF-8.

Flussi di log

Un flusso di log è una sequenza di log eventi che condividono la stessa origine. Più precisamente, un flusso di log in genere è destinato a rappresentare la sequenza di eventi provenienti dall'istanza dell'applicazione o dalla risorsa monitorata. Ad esempio, un flusso di log può essere associata a un log di accesso Apache in un determinato host. Quando non è più necessario un flusso di log, è possibile eliminarlo utilizzando il comando aws logs. delete-log-stream

Gruppi di log

I gruppi di log definiscono i gruppi dei flussi di log che condividono le stesse impostazioni di conservazione, monitoraggio e controllo degli accessi. Ogni flusso di log deve appartenere a un gruppo di log. Ad esempio, se disponi di un flusso di log separato per i log di accesso Apache da ogni host, puoi raggruppare tali flussi di log in una singola chiamata di gruppi di log `MyWebsite.com/Apache/access_log`.

Non vi è alcun limite al numero di flussi di log che possono appartenere a un gruppo di log.

Filtri di parametri

Puoi utilizzare filtri parametri per estrarre osservazioni sui parametri dagli eventi acquisiti e trasformarle in punti dati in un parametro CloudWatch . I filtri di parametro vengono assegnati ai gruppi di log e tutti i filtri assegnati a un gruppo di log vengono applicati ai relativi flussi di log.

Impostazioni della conservazione

Le impostazioni di conservazione possono essere utilizzate per specificare per quanto tempo gli eventi di registro vengono conservati nei CloudWatch registri. Gli eventi di log scaduti vengono eliminati automaticamente. Proprio come i filtri parametro, anche le impostazioni della conservazione vengono assegnate ai gruppi di log e la conservazione assegnata a un gruppo di log viene applicata ai relativi flussi di log.

Fatturazione e costi di File di log Amazon CloudWatch

Per ulteriori informazioni su come analizzare i costi e l'utilizzo di File di log CloudWatch e CloudWatch e per le best practice su come ridurre i costi, consulta la sezione CloudWatch billing and cost.

Per ulteriori informazioni sui prezzi, consulta Prezzi di Amazon CloudWatch.

Quando effettui la registrazione ad AWS, puoi iniziare a utilizzare CloudWatch Logs gratuitamente tramite il piano gratuito AWS.

Si applicano le tariffe standard per i log archiviati da altri servizi che utilizzano File di log CloudWatch, ad esempio i flussi di log Amazon VPC e i log Lambda.

Guida introduttiva ai CloudWatch registri

Per raccogliere i log dalle istanze Amazon EC2 e dai server locali CloudWatch in Logs, utilizza l'agente unificato. CloudWatch Consente di raccogliere sia i log che i parametri avanzati con un agente. Offre il supporto tra sistemi operativi, inclusi i server che eseguono Windows Server. Questo agente fornisce inoltre prestazioni migliori.

Se utilizzi l'agente unificato per raccogliere i parametri, abilita la raccolta di CloudWatch parametri di sistema aggiuntivi, per la visibilità degli ospiti. Inoltre, supporta la raccolta dei parametri personalizzati utilizzando `StatsD` o `collectd`.

Per ulteriori informazioni, consulta Installazione dell' CloudWatch agente nella Amazon CloudWatch User Guide.

Il vecchio agente CloudWatch Logs, che supporta solo la raccolta di log dai server che eseguono Linux, è obsoleto e non è più supportato. Per informazioni sulla migrazione dal vecchio agente CloudWatch Logs all'agente unificato, consulta Creare il file di configurazione dell'agente con la procedura guidata. CloudWatch

Indice

Prerequisiti

Per utilizzare Amazon CloudWatch Logs è necessario un AWS account. Il tuo AWS account ti consente di utilizzare servizi (ad esempio Amazon EC2) per generare log che puoi visualizzare nella CloudWatch console, un'interfaccia basata sul Web. Inoltre, puoi installare e configurare il file (). AWS Command Line Interface AWS CLI

Registrati per un Account AWS

Se non ne hai uno Account AWS, completa i seguenti passaggi per crearne uno.

Per iscriverti a un Account AWS

1. Apri la pagina all'indirizzo https://portal.aws.amazon.com/billing/signup.
2. Segui le istruzioni online.

   Nel corso della procedura di registrazione riceverai una telefonata, durante la quale sarà necessario inserire un codice di verifica attraverso la tastiera del telefono.

   Quando ti iscrivi a un Account AWS, Utente root dell'account AWSviene creato un. L'utente root dispone dell'accesso a tutte le risorse e tutti i Servizi AWS nell'account. Come best practice di sicurezza, assegna l'accesso amministrativo a un utente amministrativo e utilizza solo l'utente root per eseguire attività che richiedono l'accesso di un utente root.

AWS ti invia un'email di conferma dopo il completamento della procedura di registrazione. È possibile visualizzare l'attività corrente dell'account e gestire l'account in qualsiasi momento accedendo all'indirizzo https://aws.amazon.com/ e selezionando Il mio account.

Creazione di un utente amministratore

Dopo esserti registrato a Account AWS, crea un utente amministrativo in modo da non utilizzare l'utente root per le attività quotidiane.

Proteggi i tuoi Utente root dell'account AWS

1. Accedi AWS Management Consolecome proprietario dell'account scegliendo Utente root e inserendo il tuo indirizzo Account AWS email. Nella pagina successiva, inserisci la password.

   Per informazioni sull'accesso utilizzando un utente root, consulta la pagina Accesso come utente root della Guida per l'utente diAccedi ad AWS .
2. Abilita l'autenticazione a più fattori (MFA) per l'utente root.

   Per istruzioni, consulta Abilitare un dispositivo MFA virtuale per l'utente Account AWS root (console) nella Guida per l'utente IAM.

Creazione di un utente amministratore

- Per le tue attività amministrative quotidiane, assegna l'accesso amministrativo a un utente amministratore in AWS IAM Identity Center.

Per ricevere istruzioni, consulta Nozioni di base nella Guida per l'utente diAWS IAM Identity Center .

Accesso come utente amministratore

- Per accedere con l'utente IAM Identity Center, utilizza l'URL di accesso che è stato inviato al tuo indirizzo e-mail quando hai creato l'utente IAM Identity Center.

  Per informazioni sull'accesso utilizzando un utente IAM Identity Center, consulta AWS Accedere al portale di accesso nella Guida per l'Accedi ad AWS utente.

Impostazione dell'interfaccia a riga di comando

È possibile utilizzare il AWS CLI per eseguire operazioni di CloudWatch registro.

Per informazioni su come installare e configurare AWS CLI, vedere Getting Set Up with the AWS Command Line Interface nella Guida per l'AWS Command Line Interface utente.

Usa l' CloudWatch agente unificato per iniziare a usare Logs CloudWatch

Per ulteriori informazioni sull'utilizzo dell' CloudWatch agente unificato per iniziare a usare i log, consulta Collect Metrics and CloudWatch Logs from Amazon EC2 Instances and On-Premises Servers with the Agent nella Amazon User Guide. CloudWatch CloudWatch Per installare, configurare e avviare l'agente, completa la procedura di seguito riportata. Se non utilizzi l'agente per raccogliere anche i parametri, puoi ignorare tutte le sezioni che fanno riferimento ai CloudWatch parametri.

Se attualmente utilizzi il vecchio agente CloudWatch Logs e desideri passare al nuovo agente unificato, ti consigliamo di utilizzare la procedura guidata inclusa nel nuovo pacchetto dell'agente. Questa procedura guidata può leggere il file di configurazione corrente dell'agente CloudWatch Logs e configurare l' CloudWatchagente per raccogliere gli stessi registri. Per ulteriori informazioni sulla procedura guidata, consulta Create the CloudWatch Agent Configuration File with the Wizard nella Amazon CloudWatch User Guide.

Usa l' CloudWatch agente precedente per iniziare a usare Logs CloudWatch

> **Important**
>
> CloudWatch include un CloudWatch agente unificato in grado di raccogliere sia log che metriche dalle istanze EC2 e dai server locali. Il vecchio agente che utilizza solo i log è obsoleto e non è più supportato.
> Per informazioni sulla migrazione dal vecchio agente logs-only all'agente unificato, consulta Creare il file di configurazione dell'agente con la procedura guidata. CloudWatch
> Il resto di questa sezione spiega l'uso del vecchio agente CloudWatch Logs per i clienti che lo utilizzano ancora.

Utilizzando l'agente CloudWatch Logs, puoi pubblicare i dati di registro dalle istanze Amazon EC2 che eseguono Linux o Windows Server e gli eventi registrati da. AWS CloudTrailTi consigliamo invece di utilizzare l'agente CloudWatch unificato per pubblicare i dati di registro. Per ulteriori informazioni sul nuovo agente, consulta Collect metrics and logs from Amazon EC2 Instances and On-Premises Servers with the CloudWatch Agent nella Amazon User Guide. CloudWatch

Indice

CloudWatch Registra i prerequisiti dell'agente

L'agente CloudWatch Logs richiede la versione Python 2.7, 3.0 o 3.3 e una delle seguenti versioni di Linux:

- Amazon Linux 2014.03.02 o versioni successive. Amazon Linux 2 non è supportato
- Server Ubuntu versione 12.04, 14.04 o 16.04
- CentOS versione 6, 6.3, 6.4, 6.5 o 7.0
- Red Hat Enterprise Linux (RHEL) versione 6.5 o 7.0
- Debian 8.0

Avvio rapido: installa e configura l'agente CloudWatch Logs su un'istanza Linux EC2 in esecuzione

Important

Il vecchio logs agent è obsoleto. CloudWatch include un agente unificato in grado di raccogliere sia i log che le metriche dalle istanze EC2 e dai server locali. Per ulteriori informazioni, consulta Guida introduttiva ai CloudWatch registri.
Per informazioni sulla migrazione dal vecchio agente CloudWatch Logs all'agente unificato, consulta Creare il file di configurazione dell'agente con la procedura guidata. CloudWatch
L'agente di log più vecchio supporta solo le versioni da 2.6 a 3.5 di Python. Inoltre, il vecchio agente CloudWatch Logs non supporta Instance Metadata Service versione 2 (IMDSv2). Se il server utilizza IMDSv2, è necessario utilizzare l'agente unificato più recente anziché il vecchio agente Logs. CloudWatch
Il resto di questa sezione spiega l'uso del vecchio agente CloudWatch Logs per i clienti che lo utilizzano ancora.

Tip

CloudWatch include un nuovo agente unificato in grado di raccogliere sia i log che le metriche dalle istanze EC2 e dai server locali. Se non utilizzi già il vecchio agente CloudWatch Logs, ti consigliamo di utilizzare il nuovo agente unificato. CloudWatch Per ulteriori informazioni, consulta Guida introduttiva ai CloudWatch registri.

Inoltre, il vecchio agente non supporta Instance Metadata Service versione 2 (IMDSv2). Se il server utilizza IMDSv2, è necessario utilizzare l'agente unificato più recente anziché il vecchio agente Logs. CloudWatch
Il resto di questa sezione spiega l'uso del vecchio agente Logs. CloudWatch

Configura il vecchio agente CloudWatch Logs su un'istanza EC2 Linux in esecuzione

Puoi utilizzare il programma di installazione dell'agente CloudWatch Logs su un'istanza EC2 esistente per installare e configurare l'agente Logs. CloudWatch Dopo aver completato l'installazione, i log cominciano a fluire automaticamente dall'istanza al flusso di log creato durante l'installazione dell'agente. L'agente conferma l'avvio e continua l'esecuzione fino a quando non lo disabiliterai.

Oltre a utilizzare l'agente, puoi anche pubblicare i dati di registro utilizzando CloudWatch Logs SDK o l' AWS CLIAPI Logs. CloudWatch AWS CLI È più adatto per la pubblicazione di dati sulla riga di comando o tramite script. L'SDK CloudWatch Logs è più adatto per pubblicare i dati di registro direttamente dalle applicazioni o per creare un'applicazione di pubblicazione dei log personalizzata.

Passaggio 1: configura il tuo ruolo o utente IAM per Logs CloudWatch

L'agente CloudWatch Logs supporta i ruoli e gli utenti IAM. Se l'istanza dispone già di un ruolo IAM a essa associato, assicurati di includere la policy IAM sottostante. Se non disponi già di un ruolo IAM assegnato alla tua istanza, puoi utilizzare le tue credenziali IAM per le fasi successive o puoi assegnare un ruolo IAM a tale istanza. Per ulteriori informazioni, consulta la sezione relativa al collegamento di un ruolo IAM a un'istanza.

Per configurare il tuo ruolo o utente IAM per Logs CloudWatch

1. Aprire la console IAM all'indirizzo https://console.aws.amazon.com/iam/.
2. Nel riquadro di navigazione, seleziona Ruoli.
3. Scegliere il ruolo selezionando il nome del ruolo (non selezionare la casella di controllo accanto al nome).
4. Scegliere Attach Policies (Collega policy), Create Policy (Crea policy).

   Verrà aperta una nuova scheda o finestra del browser.
5. Scegliere la scheda JSON e digitare il documento di policy JSON seguente.

```
{
```

```
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents",
        "logs:DescribeLogStreams"
    ],
      "Resource": [
        "*"
    ]
  }
 ]
}
```

6. Al termine, selezionare Review policy (Rivedi policy). In Policy Validator (Validatore di policy) vengono segnalati eventuali errori di sintassi.
7. Nella pagina Review policy (Rivedi policy) digitare i valori per Name (Nome) e Description (Descrizione) (facoltativa) per la policy che si sta creando. Consulta il Summary (Riepilogo) della policy per visualizzare le autorizzazioni concesse dalla policy. Seleziona Create policy (Crea policy) per salvare il proprio lavoro.
8. Chiudere la scheda o la finestra del browser e tornare alla pagina Add permissions (Aggiungi autorizzazioni) per il ruolo. Scegliere Refresh (Aggiorna) e quindi scegliere la nuova policy per collegarla al ruolo.
9. Scegli Attach Policy (Collega policy).

Fase 2: installare e configurare CloudWatch i log su un'istanza Amazon EC2 esistente

Il processo di installazione dell'agente CloudWatch Logs varia a seconda che l'istanza Amazon EC2 esegua Amazon Linux, Ubuntu, CentOS o Red Hat. Utilizza la procedura appropriata per la versione di Linux della tua istanza.

Per installare e configurare CloudWatch i log su un'istanza Amazon Linux esistente

A partire dall'AMI Amazon Linux 2014.09, l'agente CloudWatch Logs è disponibile come installazione RPM con il pacchetto awslogs. Le versioni precedenti di Amazon Linux possono accedere al pacchetto awslogs aggiornando l'istanza con il comando `sudo yum update -y`. Installando il pacchetto awslogs come RPM anziché utilizzare il programma di installazione CloudWatch Logs,

l'istanza riceve aggiornamenti e patch regolari dei pacchetti senza dover reinstallare manualmente l'agente Logs. AWS CloudWatch

> **Warning**
>
> Non aggiornate l'agente CloudWatch Logs utilizzando il metodo di installazione RPM se in precedenza avete utilizzato lo script Python per installare l'agente. Ciò potrebbe causare problemi di configurazione che impediscono all'agente CloudWatch Logs di inviare i log a. CloudWatch

1. Connessione a un'istanza Amazon Linux. Per ulteriori informazioni, consulta Conessione all'istanza nella Guida per l'utente di Amazon EC2 per le istanze Linux.

   Per ulteriori informazioni sui problemi di connessione, consulta Risoluzione dei problemi di connessione all'istanza nella Guida per l'utente di Amazon EC2 per le istanze Linux.

2. Aggiornare l'istanza Amazon Linux per prelevare le modifiche più recenti nei repository del pacchetto.

   ```
   sudo yum update -y
   ```

3. Installare il pacchetto `awslogs`. Questo è il metodo consigliato per l'installazione di awslogs nelle istanze Amazon Linux.

   ```
   sudo yum install -y awslogs
   ```

4. Modificare il file `/etc/awslogs/awslogs.conf` per configurare i log da monitorare. Per ulteriori informazioni sulla modifica di questo file, consulta Riferimento dell'agente CloudWatch Logs.

5. Per impostazione predefinita, `/etc/awslogs/awscli.conf` punta alla Regione us-east-1. Per portare i log in un'altra Regione, modificare il file `awscli.conf` e specificare la Regione.

6. Avviare il servizio `awslogs`.

   ```
   sudo service awslogs start
   ```

   Se si esegue Amazon Linux 2, avviare il servizio `awslogs` con il comando seguente.

```
sudo systemctl start awslogsd
```

7. Controlla che il file `/var/log/awslogs.log` non contenga errori registrati all'avvio del servizio (facoltativo).
8. Esegui il comando seguente per avviare il servizio `awslogs` a ogni avvio del sistema (facoltativo).

   ```
   sudo chkconfig awslogs on
   ```

   Se si esegue Amazon Linux 2, utilizzare il comando seguente per avviare il servizio a ogni avvio del sistema.

   ```
   sudo systemctl enable awslogsd.service
   ```

9. Il gruppo di log e il flusso di log appena creati dovrebbero venire visualizzati nella console CloudWatch dopo qualche minuto di esecuzione dell'agente.

   Per ulteriori informazioni, consulta Visualizza i dati di registro inviati ai registri CloudWatch .

Per installare e configurare CloudWatch i log su un'istanza esistente di Ubuntu Server, CentOS o Red Hat

Se utilizzi un'AMI che esegue Ubuntu Server, CentOS o Red Hat, utilizza la seguente procedura per installare manualmente l'agente CloudWatch Logs sulla tua istanza.

1. Connettiti all'istanza EC2. Per ulteriori informazioni, consulta Conessione all'istanza nella Guida per l'utente di Amazon EC2 per le istanze Linux.

   Per ulteriori informazioni sui problemi di connessione, consulta Risoluzione dei problemi di connessione all'istanza nella Guida per l'utente di Amazon EC2 per le istanze Linux.
2. Esegui il programma di installazione dell'agente CloudWatch Logs utilizzando una delle due opzioni. È possibile eseguirlo direttamente da Internet o scaricare i file ed eseguirlo autonomamente.

   > **Note**
   >
   > Se si esegue CentOS 6.x o Red Hat 6.x o Ubuntu 12.04, utilizzare la procedura di download ed esecuzione del programma di installazione in maniera autonoma.

L'installazione dell'agente CloudWatch Logs direttamente da Internet non è supportata su questi sistemi.

> **Note**
>
> Su Ubuntu, esegui `apt-get update` prima di eseguire i comandi sottostanti.

Per eseguirlo direttamente da Internet, utilizza i comandi seguenti e segui le istruzioni:

```
curl https://s3.amazonaws.com/aws-cloudwatch/downloads/latest/awslogs-agent-
setup.py -O
```

```
sudo python ./awslogs-agent-setup.py --region us-east-1
```

Se il comando precedente non funziona, prova il seguente:

```
sudo python3 ./awslogs-agent-setup.py --region us-east-1
```

Per scaricarlo ed eseguirlo autonomamente, utilizza i comandi seguenti e segui le istruzioni:

```
curl https://s3.amazonaws.com/aws-cloudwatch/downloads/latest/awslogs-agent-
setup.py -O
```

```
curl https://s3.amazonaws.com/aws-cloudwatch/downloads/latest/
AgentDependencies.tar.gz -O
```

```
tar xvf AgentDependencies.tar.gz -C /tmp/
```

```
sudo python ./awslogs-agent-setup.py --region us-east-1 --dependency-path /tmp/
AgentDependencies
```

È possibile installare l'agente CloudWatch Logs specificando le regioni us-east-1, us-west-1, us-west-2, ap-south-1, ap-northeast-2, ap-southeast-1, ap-southeast-2, ap-northeast-1, ap-northeast-1, eu-central-1, eu-west-1 o sa-east-1.

> **Note**
>
> Per ulteriori informazioni sulla versione attuale e sulla cronologia delle versioni di `awslogs-agent-setup`, consulta CHANGELOG.txt.

Il programma di installazione dell'agente Logs richiede determinate informazioni durante la configurazione. CloudWatch Prima di iniziare, è necessario sapere quale file di log monitorare e il suo formato timestamp. È inoltre necessario disporre delle informazioni seguenti.

Elemento	Descrizione
AWS ID della chiave di accesso	Premi invio se utilizzi un ruolo IAM. Altrimenti, inserisci l'ID della tua chiave di AWS accesso.
AWS chiave di accesso segreta	Premi invio se utilizzi un ruolo IAM. Altrimenti, inserisci la tua chiave di accesso AWS segreta.
Nome della Regione predefinito	Premere Invio. La Regione di default è us-east-2. Si può impostare questo a us-east-1, us-west-1, us-west-2, ap-south-1, ap-northe ast-2, ap-southeast-1, ap-southeast-2, ap-northeast-1, eu-central-1, eu-west-1, o sa-east-1.
Formato di output predefinito	Lascia questo campo vuoto e premi invio.
Percorso del file di log da caricare	Posizione del file che contiene i dati di log da inviare. Il programma di installazione ti suggerisce un percorso.
Nome del gruppo di log di destinazione	Nome del gruppo di log. Il programma di installazione ti suggerisce un nome per il gruppo di log.
Nome del flusso di log di destinazione	Per impostazione predefinita, questo è il nome dell'host. Il programma di installazione ti suggerisce un nome host.

Elemento	Descrizione
Formato timestamp	Specifica il formato del timestamp all'interno del file di log specifica to. Scegli il valore di personalizzazione per specificare il tuo formato.
Posizione iniziale	In che modo i dati vengono caricati. Imposta questo valore su start_of_file per caricare tutto il contenuto del file. Imposta su end_of_file per caricare solo i dati appena aggiunti.

Dopo aver completato questa procedura, il programma di installazione chiede se intendi configurare un altro file di log. Puoi eseguire il processo quante volte lo desideri per ogni file di log. Se non hai altri file di log da monitorare, scegli N quando richiesto dal programma di installazione per configurare un altro log. Per ulteriori informazioni sulle impostazioni all'interno del file di configurazione dell'agente, consulta Riferimento dell'agente CloudWatch Logs.

> **Note**
>
> La configurazione di più origini di log per l'invio di dati a un unico flusso di log non è supportata.

3. Il gruppo di log e il flusso di log appena creati dovrebbero venire visualizzati nella console CloudWatch dopo qualche minuto di esecuzione dell'agente.

   Per ulteriori informazioni, consulta Visualizza i dati di registro inviati ai registri CloudWatch .

Avvio rapido: installa e configura l'agente CloudWatch Logs su un'istanza Linux EC2 al momento del lancio

> **Tip**
>
> Il vecchio agente CloudWatch Logs discusso in questa sezione sta per diventare obsoleto. Ti consigliamo vivamente di utilizzare invece il nuovo CloudWatch agente unificato in grado di raccogliere sia i log che le metriche. Inoltre, il vecchio agente CloudWatch Logs richiede Python 3.3 o versioni precedenti e queste versioni non sono installate su nuove istanze

EC2 per impostazione predefinita. Per ulteriori informazioni sull'agente unificato, consulta Installazione CloudWatch dell'agente. CloudWatch
Il resto di questa sezione spiega l'uso del vecchio agente CloudWatch Logs.

Installazione del vecchio agente CloudWatch Logs su un'istanza EC2 Linux all'avvio

Puoi utilizzare i dati utente di Amazon EC2, una funzionalità di Amazon EC2 che consente il trasferimento di informazioni parametriche all'istanza all'avvio, per installare e configurare CloudWatch l'agente Logs su quell'istanza. Per trasmettere le informazioni di installazione e configurazione dell'agente CloudWatch Logs ad Amazon EC2, puoi fornire il file di configurazione in una posizione di rete, ad esempio un bucket Amazon S3.

La configurazione di più origini di log per l'invio di dati a un unico flusso di log non è supportata.

Prerequisito

Crea un file di configurazione dell'agente che descriva tutti i gruppi di log e i flussi di log. Si tratta di un file di testo che descrive i file di log da monitorare e i gruppi di log e i flussi di log sui quali caricarli. L'agente utilizza questo file di configurazione e inizia a monitore e a caricare tutti i file di log in esso descritti. Per ulteriori informazioni sulle impostazioni all'interno del file di configurazione dell'agente, consulta Riferimento dell'agente CloudWatch Logs.

Di seguito viene illustrato un esempio di file di configurazione dell'agente per Amazon Linux 2

```
[general]
state_file = /var/lib/awslogs/state/agent-state

[/var/log/messages]
file = /var/log/messages
log_group_name = /var/log/messages
log_stream_name = {instance_id}
datetime_format = %b %d %H:%M:%S
```

Di seguito un esempio di file di configurazione dell'agente per Ubuntu

```
[general]
state_file = /var/awslogs/state/agent-state

[/var/log/syslog]
```

```
file = /var/log/syslog
log_group_name = /var/log/syslog
log_stream_name = {instance_id}
datetime_format = %b %d %H:%M:%S
```

Configurazione di un ruolo IAM

1. Aprire la console IAM all'indirizzo https://console.aws.amazon.com/iam/.
2. Nel riquadro di navigazione scegli Policies (Policy), quindi Create Policy (Crea policy).
3. Nella pagina Create Policy (Crea policy), in Create Your Own Policy (Crea la tua policy), scegli Select (Seleziona). Per ulteriori informazioni sulla creazione di policy personalizzate, consulta Policy IAM per Amazon EC2 nella Guida per l'utente di Amazon EC2 per le istanze Linux.
4. Nella pagina Review Policy (Rivedi policy), in Policy Name (Nome policy) digita un nome per la policy.
5. In Policy Document (Documento policy), copia la policy seguente:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogGroup",
                "logs:CreateLogStream",
                "logs:PutLogEvents",
                "logs:DescribeLogStreams"
            ],
            "Resource": [
                "arn:aws:logs:*:*:*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::myawsbucket/*"
            ]
        }
    ]
```

```
}
```

6. Scegliere Create Policy (Crea policy).
7. Nel riquadro di navigazione selezionare Roles (Ruoli), quindi Create new role (Crea nuovo ruolo).
8. Nella pagina Set Role Name (Imposta nome ruolo), digita un nome per il ruolo e scegli Next Step (Fase successiva).
9. Nella pagina Select Role Type (Seleziona tipo di ruolo), scegli Select (Seleziona) accanto ad Amazon EC2.
10. Nella pagina Attach Policy (Collega policy), nell'intestazione della tabella, scegli Policy Type (Tipo di policy), Customer Managed (Gestito dal cliente).
11. Seleziona la policy IAM creata, quindi scegli Next Step (Fase successiva).
12. Selezionare Crea ruolo.

    Per ulteriori informazioni sugli utenti e sulle policy, consulta Utenti e gruppi IAM e Gestione delle policy IAM nella Guida per l'utente IAM.

Per avviare una nuova istanza e abilitare Logs CloudWatch

1. Aprire la console Amazon EC2 all'indirizzo https://console.aws.amazon.com/ec2/.
2. Scegliere Launch Instance (Avvia istanza).

   Per ulteriori informazioni, consulta Avvio di un'istanza nella Guida per l'utente di Amazon EC2 per istanze Linux.
3. Nella pagina Step 1: Choose an Amazon Machine Image (AMI) (Fase 1: scegli un'Amazon Machine Image (AMI)), seleziona il tipo di istanza Linux da avviare, quindi, nella pagina Step 2: Choose an Instance Type (Fase 2: scegli un tipo di istanza), scegli Next: Configure Instance Details (Successivo: configura i dettagli dell'istanza).

   Assicurati che cloud-init sia incluso nell'Amazon Machine Image (AMI). Le AMI Amazon Linux e le AMI per Ubuntu e RHEL includono già cloud-init, ma CentOS e altre AMI potrebbero non esserlo. Marketplace AWS
4. Nella pagina Step 3: Configure Instance Details (Fase 3: configura i dettagli dell'istanza), in IAM role (Ruolo IAM), seleziona il ruolo IAM creato.
5. Sotto a Advanced Details (Dettagli avanzati), in User data (Dati utente), incolla il seguente script nella casella. In seguito aggiorna tale script cambiando il valore dell'opzione -c nella posizione del file di configurazione dell'agente.

```
#!/bin/bash
curl https://s3.amazonaws.com/aws-cloudwatch/downloads/latest/awslogs-agent-
setup.py -O
chmod +x ./awslogs-agent-setup.py
./awslogs-agent-setup.py -n -r us-east-1 -c s3://DOC-EXAMPLE-BUCKET1/my-config-file
```

6. Apporta tutte le altre modifiche all'istanza, controlla le impostazioni di avvio, quindi scegli Launch (Avvia).
7. Dovresti vedere il gruppo di log e il flusso di log appena creati nella CloudWatch console dopo che l'agente è stato in esecuzione per alcuni istanti.

   Per ulteriori informazioni, consulta Visualizza i dati di registro inviati ai registri CloudWatch .

Avvio rapido: consenti alle istanze Amazon EC2 che eseguono Windows Server 2016 di inviare log a Logs utilizzando l'agente CloudWatch Logs CloudWatch

> **Tip**
>
> CloudWatch include un nuovo agente unificato in grado di raccogliere sia i log che le metriche dalle istanze EC2 e dai server locali. Ti consigliamo di utilizzare il nuovo agente CloudWatch unificato. Per ulteriori informazioni, consulta Guida introduttiva ai CloudWatch registri.
> Il resto di questa sezione spiega l'uso del vecchio agente Logs. CloudWatch

Consenti alle istanze Amazon EC2 che eseguono Windows Server 2016 di inviare log a Logs utilizzando il vecchio CloudWatch agente Logs CloudWatch

È possibile utilizzare diversi metodi per consentire alle istanze che eseguono Windows Server 2016 di inviare i log ai registri. CloudWatch Nella procedura di questa sezione viene utilizzato Run Command di Systems Manager. Per informazioni sugli altri metodi possibili, consulta Invio di registri, eventi e contatori delle prestazioni ad Amazon. CloudWatch

Fasi

- Download del file di configurazione di esempio
- Configura il file JSON per CloudWatch

- Creare un ruolo IAM per Systems Manager
- Verifica dei prerequisiti di Systems Manager
- Verifica dell'accesso a Internet
- Abilita CloudWatch i log utilizzando il comando Systems Manager Run

Download del file di configurazione di esempio

Download il seguente file di esempio su computer: `AWS.EC2.Windows.CloudWatch.json`.

Configura il file JSON per CloudWatch

È possibile determinare a quali log inviare i dati CloudWatch specificando le proprie scelte in un file di configurazione. Il processo di creazione di questo file e l'indicazione delle scelte possono richiedere 30 minuti o più per il completamento. Una volta completata questa attività una volta, puoi riutilizzare il file di configurazione su tutte le istanze.

Fasi

- Passaggio 1: abilitare i registri CloudWatch
- Fase 2: Configurare le impostazioni per CloudWatch
- Fase 3: configurazione dei dati da inviare
- Fase 4: configurazione del controllo di flusso
- Fase 5: salvataggio del contenuto JSON

Passaggio 1: abilitare i registri CloudWatch

Nella parte superiore del file JSON, modifica "falso" in "vero" per `IsEnabled`:

```
"IsEnabled": true,
```

Fase 2: Configurare le impostazioni per CloudWatch

Specifica le credenziali, la Regione, un nome per il gruppo di log e uno spazio dei nomi del flusso di log. Ciò consente all'istanza di inviare i dati di registro ai CloudWatch registri. Per inviare gli stessi dati di registro a posizioni diverse, puoi aggiungere sezioni aggiuntive con ID univoci (ad esempio, "CloudWatchLogs2" e CloudWatchLogs 3") e una regione diversa per ogni ID.

Per configurare le impostazioni per l'invio dei dati di registro ai registri CloudWatch

1. Nel file JSON, individuare la sezione `CloudWatchLogs`.

```
{
    "Id": "CloudWatchLogs",
    "FullName":
 "AWS.EC2.Windows.CloudWatch.CloudWatchLogsOutput,AWS.EC2.Windows.CloudWatch",
    "Parameters": {
        "AccessKey": "",
        "SecretKey": "",
        "Region": "us-east-1",
        "LogGroup": "Default-Log-Group",
        "LogStream": "{instance_id}"
    }
},
```

2. Lasciare vuoti i campi `AccessKey` e `SecretKey`. Puoi configurare le credenziali tramite un ruolo IAM.
3. In `Region` digitare la Regione alla quale inviare dati di log (ad esempio, `us-east-2`).
4. Per `LogGroup` digita il nome del gruppo di log. Questo nome viene visualizzato nella schermata Log Groups (Gruppi di log) nella console CloudWatch.
5. Per `LogStream` digita il flusso di log di destinazione. Questo nome viene visualizzato nella schermata Log Groups > Streams della CloudWatch console.

   Se si utilizza `{instance_id}`, il nome del flusso di log predefinito è l'ID istanza di tale istanza.

   Se specifichi un nome di log stream che non esiste già, CloudWatch Logs lo crea automaticamente per te. È possibile definire un nome per il flusso di log utilizzando una stringa letterale, le variabili predefinite `{instance_id}`, `{hostname}` e `{ip_address}` oppure una combinazione di queste.

Fase 3: configurazione dei dati da inviare

È possibile inviare i dati del registro degli eventi, i dati di Event Tracing for Windows (ETW) e altri dati di registro ai registri. CloudWatch

Per inviare i dati del registro degli eventi dell'applicazione Windows a Logs CloudWatch

1. Nel file JSON, individuare la sezione `ApplicationEventLog`.

```
{
    "Id": "ApplicationEventLog",
    "FullName":
 "AWS.EC2.Windows.CloudWatch.EventLog.EventLogInputComponent,AWS.EC2.Windows.CloudWatch",
    "Parameters": {
        "LogName": "Application",
        "Levels": "1"
    }
},
```

2. Per `Levels`, specifica il tipo di messaggio da caricare. È possibile specificare uno dei seguenti valori:

   - **1**: vengono caricati solo i messaggi di errore.
   - **2**: vengono caricati solo i messaggi di avviso.
   - **4**: vengono caricati solo i messaggi informativi.

   È possibile combinare i valori per includere diversi tipi di messaggio. Per esempio, il valore **3** carica ad esempio i messaggi di errore (**1**) e i messaggi di avviso (**2**). Il valore **7** carica i messaggi di errore (**1**), i messaggi di avviso (**2**) e i messaggi informativi (**4**).

Per inviare i dati del registro di sicurezza ai CloudWatch registri

1. Nel file JSON, individuare la sezione `SecurityEventLog`.

```
{
    "Id": "SecurityEventLog",
    "FullName":
 "AWS.EC2.Windows.CloudWatch.EventLog.EventLogInputComponent,AWS.EC2.Windows.CloudWatch",
    "Parameters": {
        "LogName": "Security",
        "Levels": "7"
    }
},
```

2. Per `Levels`, digita **7** per caricare tutti i messaggi.

Per inviare i dati del registro degli eventi di sistema a CloudWatch Logs

1. Nel file JSON, individuare la sezione `SystemEventLog`.

```
{
    "Id": "SystemEventLog",
    "FullName":
 "AWS.EC2.Windows.CloudWatch.EventLog.EventLogInputComponent,AWS.EC2.Windows.CloudWatch",
    "Parameters": {
        "LogName": "System",
        "Levels": "7"
    }
},
```

2. Per `Levels`, specifica il tipo di messaggio da caricare. È possibile specificare uno dei seguenti valori:

   - **1**: vengono caricati solo i messaggi di errore.
   - **2**: vengono caricati solo i messaggi di avviso.
   - **4**: vengono caricati solo i messaggi informativi.

   È possibile combinare i valori per includere diversi tipi di messaggio. Per esempio, il valore **3** carica ad esempio i messaggi di errore (**1**) e i messaggi di avviso (**2**). Il valore **7** carica i messaggi di errore (**1**), i messaggi di avviso (**2**) e i messaggi informativi (**4**).

Per inviare altri tipi di dati del registro degli eventi a CloudWatch Logs

1. Nel file JSON, aggiungere una nuova sezione. Ogni sezione deve avere un `Id` univoco.

```
{
    "Id": "Id-name",
    "FullName":
 "AWS.EC2.Windows.CloudWatch.EventLog.EventLogInputComponent,AWS.EC2.Windows.CloudWatch",
    "Parameters": {
        "LogName": "Log-name",
        "Levels": "7"
    }
},
```

2. Per `Id` digitare un nome per il log da caricare (ad esempio, **`WindowsBackup`**).

3. Per `LogName`, digita il nome del log da caricare. È possibile trovare il nome del log nel modo seguente.

   a. Aprire il visualizzatore di eventi.

   b. Nel riquadro di navigazione, selezionare Applications and Services Logs (Log di applicazioni e servizi).

   c. Andare al log, quindi scegliere Actions (Operazioni) e Properties (Proprietà).

4. Per `Levels`, specifica il tipo di messaggio da caricare. È possibile specificare uno dei seguenti valori:

   - **1**: vengono caricati solo i messaggi di errore.
   - **2**: vengono caricati solo i messaggi di avviso.
   - **4**: vengono caricati solo i messaggi informativi.

   È possibile combinare i valori per includere diversi tipi di messaggio. Per esempio, il valore **3** carica ad esempio i messaggi di errore (**1**) e i messaggi di avviso (**2**). Il valore **7** carica i messaggi di errore (**1**), i messaggi di avviso (**2**) e i messaggi informativi (**4**).

Per inviare i dati di Event Tracing for Windows ai registri CloudWatch

ETW (traccia eventi per Windows) fornisce meccanismi di log efficienti e dettagliati ai quali le applicazioni possono scrivere log. Ogni ETW è controllato da un gestore di sessione che può avviare e terminare la sessione di log. Ogni sessione dispone di un fornitore e di uno o più consumatori.

1. Nel file JSON, individuare la sezione `ETW`.

```
{
    "Id": "ETW",
    "FullName":
 "AWS.EC2.Windows.CloudWatch.EventLog.EventLogInputComponent,AWS.EC2.Windows.CloudWatch",
    "Parameters": {
        "LogName": "Microsoft-Windows-WinINet/Analytic",
        "Levels": "7"
    }
},
```

2. Per `LogName`, digita il nome del log da caricare.

3. Per `Levels`, specifica il tipo di messaggio da caricare. È possibile specificare uno dei seguenti valori:

   - **1**: vengono caricati solo i messaggi di errore.
   - **2**: vengono caricati solo i messaggi di avviso.
   - **4**: vengono caricati solo i messaggi informativi.

   È possibile combinare i valori per includere diversi tipi di messaggio. Per esempio, il valore **3** carica ad esempio i messaggi di errore (**1**) e i messaggi di avviso (**2**). Il valore **7** carica i messaggi di errore (**1**), i messaggi di avviso (**2**) e i messaggi informativi (**4**).

Per inviare registri personalizzati (qualsiasi file di registro basato su testo) a Logs CloudWatch

1. Nel file JSON, individuare la sezione `CustomLogs`.

```
{
    "Id": "CustomLogs",
    "FullName":
 "AWS.EC2.Windows.CloudWatch.CustomLog.CustomLogInputComponent,AWS.EC2.Windows.CloudWatch",
    "Parameters": {
        "LogDirectoryPath": "C:\\CustomLogs\\",
        "TimestampFormat": "MM/dd/yyyy HH:mm:ss",
        "Encoding": "UTF-8",
        "Filter": "",
        "CultureName": "en-US",
        "TimeZoneKind": "Local",
        "LineCount": "5"
    }
},
```

2. Per `LogDirectoryPath`, digita il percorso in cui sono memorizzati i log nell'istanza.
3. In `TimestampFormat`, digita il formato timestamp da utilizzare. Per ulteriori informazioni sui valori supportati, consulta l'argomento relativo alle stringhe di formato di data e ora personalizzate su MSDN.

> **Important**
>
> Il file di log di origine deve contenere il timestamp all'inizio di ogni riga di log e uno spazio dopo il timestamp.

4. in `Encoding`, digita la codifica del file da utilizzare (ad esempio, UTF-8). Per un elenco dei valori supportati, consulta l'argomento relativo alla classe di codifica su MSDN.

   > **Note**
   >
   > Utilizza il nome di codifica, non il nome di visualizzazione.

5. (Facoltativo) Per `Filter`, digitare il prefisso dei nomi di log. Lascia questo parametro bianco per monitorare tutti i file. Per ulteriori informazioni sui valori supportati, vedere l'argomento FileSystemWatcherFilter Proprietà su MSDN.
6. In `CultureName`, digita le impostazioni locali in cui viene registrato il timestamp (facoltativo). Se `CultureName` è vuoto, si utilizza per impostazione predefinita la stessa configurazione locale utilizzata dall'istanza di Windows. Per ulteriori informazioni relative a questo argomento, consulta la colonna `Language tag` nella tabella dell'argomento comportamento del prodotto su MSDN.

   > **Note**
   >
   > I valori `div`, `div-MV`, `hu` e `hu-HU` non sono supportati.

7. (Facoltativo) Per `TimeZoneKind`, digitare `Local` o `UTC`. Puoi impostare questo valore per fornire delle informazioni sul fuso orario se non sono incluse nel timestamp del log. Se questo parametro viene lasciato vuoto e se il timestamp non include informazioni sul fuso orario, per impostazione predefinita CloudWatch Logs utilizza il fuso orario locale. Questo parametro viene ignorato se il timestamp contiene già informazioni sul fuso orario.
8. (Facoltativo) Per `LineCount`, digitare il numero di righe dell'intestazione per identificare il file di log. Ad esempio, i file di log IIS dispongono di intestazioni praticamente identiche. È possibile immettere **5** per leggere le prime tre righe dell'intestazione del file di log per identificarlo. Nei file di log IIS, la terza riga è costituita dalla data e dal timestamp, ma non è sempre garantito che il timestamp sia diverso tra i diversi file di log. Per questo motivo, è consigliabile includere almeno una riga di dati di log reali per identificare in modo univoco il file di log.

Per inviare i dati di registro IIS a Logs CloudWatch

1. Nel file JSON, individuare la sezione `IISLog`.

```
{
    "Id": "IISLogs",
    "FullName":
 "AWS.EC2.Windows.CloudWatch.CustomLog.CustomLogInputComponent,AWS.EC2.Windows.CloudWatch",
    "Parameters": {
        "LogDirectoryPath": "C:\\inetpub\\logs\\LogFiles\\W3SVC1",
        "TimestampFormat": "yyyy-MM-dd HH:mm:ss",
        "Encoding": "UTF-8",
        "Filter": "",
        "CultureName": "en-US",
        "TimeZoneKind": "UTC",
        "LineCount": "5"
    }
},
```

2. Per `LogDirectoryPath`, digita la cartella in cui sono memorizzati i log IIS per un sito individuale (ad esempio, `C:\inetpub\logs\LogFiles\W3SVC`*n*).

> **Note**
>
> È supportato solo il formato di log W3C. I formati IIS, NCSA e quelli personalizzati non sono supportati.

3. In `TimestampFormat`, digita il formato timestamp da utilizzare. Per ulteriori informazioni sui valori supportati, consulta l'argomento relativo alle stringhe di formato di data e ora personalizzate su MSDN.
4. in `Encoding`, digita la codifica del file da utilizzare (ad esempio, UTF-8). Per ulteriori informazioni sui valori supportati, consulta l'argomento relativo alla classe di codifica su MSDN.

> **Note**
>
> Utilizza il nome di codifica, non il nome di visualizzazione.

5. (Facoltativo) Per `Filter`, digitare il prefisso dei nomi di log. Lascia questo parametro bianco per monitorare tutti i file. Per ulteriori informazioni sui valori supportati, vedere l'argomento FileSystemWatcherFilter Proprietà su MSDN.

6. In `CultureName`, digita le impostazioni locali in cui viene registrato il timestamp (facoltativo). Se `CultureName` è vuoto, si utilizza per impostazione predefinita la stessa configurazione locale utilizzata dall'istanza di Windows. Per ulteriori informazioni relative ai valori supportati, consulta la colonna `Language tag` nella tabella dell'argomento comportamento del prodotto su MSDN.

   > **Note**
   >
   > I valori `div`, `div-MV`, `hu` e `hu-HU` non sono supportati.

7. In `TimeZoneKind`, inserisci `Local` o `UTC` (facoltativo). Puoi impostare questo valore per fornire delle informazioni sul fuso orario se non sono incluse nel timestamp del log. Se questo parametro viene lasciato vuoto e se il timestamp non include informazioni sul fuso orario, per impostazione predefinita CloudWatch Logs utilizza il fuso orario locale. Questo parametro viene ignorato se il timestamp contiene già informazioni sul fuso orario.
8. (Facoltativo) Per `LineCount`, digitare il numero di righe dell'intestazione per identificare il file di log. Ad esempio, i file di log IIS dispongono di intestazioni praticamente identiche. È possibile inserire **5** per leggere le prime cinque righe dell'intestazione del file di log per identificarlo. Nei file di log IIS, la terza riga è costituita dalla data e dal timestamp, ma non è sempre garantito che il timestamp sia diverso tra i diversi file di log. Per questo motivo, ti consigliamo di includere almeno una riga di dati di log reali per identificare univocamente il file di log.

Fase 4: configurazione del controllo di flusso

Ogni tipo di dati deve avere una destinazione corrispondente nella sezione `Flows`. Ad esempio, per inviare il registro personalizzato, il registro ETW e il registro di sistema a CloudWatch Logs, aggiungete alla sezione. `(CustomLogs,ETW,SystemEventLog),CloudWatchLogs` `Flows`

> **Warning**
>
> Aggiungere una fase non valida blocca il flusso. Ad esempio, se aggiungi una fase relativa ai parametri del disco, ma la tua istanza non dispone di un disco, tutte le fasi del flusso verranno bloccate.

Puoi inviare lo stesso file di log a più destinazioni. Ad esempio, per inviare il log dell'applicazione a due destinazioni differenti, definite nella sezione `CloudWatchLogs`, aggiungi `ApplicationEventLog,(CloudWatchLogs,CloudWatchLogs2)` alla sezione `Flows`.

Per configurare il controllo di flusso

1. Nel file `AWS.EC2.Windows.CloudWatch.json`, individuare la sezione `Flows`.

```
"Flows": {
    "Flows": [
      "PerformanceCounter,CloudWatch",
      "(PerformanceCounter,PerformanceCounter2), CloudWatch2",
      "(CustomLogs, ETW, SystemEventLog),CloudWatchLogs",
      "CustomLogs, CloudWatchLogs2",
      "ApplicationEventLog,(CloudWatchLogs, CloudWatchLogs2)"
    ]
}
```

2. In `Flows`, aggiungere ciascun tipo di dati da caricare (ad esempio `ApplicationEventLog`) e la sua destinazione (ad esempio `CloudWatchLogs`).

Fase 5: salvataggio del contenuto JSON

Hai ora terminato la modifica del file JSON. Salvarlo e nella fase successiva incollare i contenuti del file in un editor di testo in un'altra finestra. I contenuti del file saranno necessari in un secondo momento nel corso di questa procedura.

Creare un ruolo IAM per Systems Manager

È necessario un ruolo IAM per ottenere le credenziali dell'istanza quando utilizzi Run Command per Systems Manager. Questo ruolo permette a Systems Manager di eseguire operazioni sull'istanza. Per ulteriori informazioni, consulta Configurazione dei ruoli di sicurezza per Systems Manager nella Guida per l'utente diAWS Systems Manager . Per informazioni su come collegare un ruolo IAM a un'istanza esistente, consulta Collegamento di un ruolo IAM a un'istanza nella Guida per l'utente di Amazon EC2 per le istanze Windows.

Verifica dei prerequisiti di Systems Manager

Prima di utilizzare Systems Manager Run Command per configurare l'integrazione con CloudWatch Logs, verificate che le istanze soddisfino i requisiti minimi. Per ulteriori informazioni, consulta Prerequisti di Systems Manager nella Guida per l'utente diAWS Systems Manager .

Verifica dell'accesso a Internet

Le tue istanze Amazon EC2 Windows Server e le istanze gestite devono disporre di accesso a Internet in uscita per poter inviare dati di log ed eventi a. CloudWatch Per ulteriori informazioni su

come configurare l'accesso a Internet, consulta Gateway Internet nella Guida per l'utente di Amazon VPC.

Abilita CloudWatch i log utilizzando il comando Systems Manager Run

Run Command ti permette di gestire la configurazione delle istanze on demand. Puoi specificare un documento Systems Manager, specificare i parametri ed eseguire il comando in una o più istanze. Il SSM Agent sull'istanza elabora il comando e configura l'istanza come specificato.

Per configurare l'integrazione con CloudWatch Logs utilizzando Run Command

1. Apri la console Amazon EC2 all'indirizzo https://console.aws.amazon.com/ec2/.
2. Aprire la console SSM all'indirizzo https://console.aws.amazon.com/systems-manager/.
3. Nel riquadro di navigazione seleziona Esegui comando.
4. Scegli Esegui comando.
5. Per il documento Command, scegli AWS- ConfigureCloudWatch.
6. Per le istanze Target, scegli le istanze da integrare con CloudWatch Logs. Se non viene visualizzata un'istanza nell'elenco, potrebbe non essere configurata per Run Command. Per ulteriori informazioni, consulta Prerequisiti di Systems Manager nella Guida per l'utente di Amazon EC2 per le istanze Windows.
7. In Status (Stato), scegliere Enabled (Abilitato).
8. In Properties (Proprietà) copiare e incollare il contenuto JSON creato nelle attività precedenti.
9. Completare i restanti campi opzionali e scegliere Esegui.

Utilizza la procedura seguente per visualizzare i risultati dell'esecuzione del comando nella console Amazon EC2.

Per visualizzare l'output del comando nella console

1. Selezionare un comando.
2. Selezionare la scheda Output.
3. Scegliere View Output (Visualizza output). La pagina di output del comando mostra i risultati di esecuzione del comando.

Avvio rapido: consenti alle istanze Amazon EC2 che eseguono Windows Server 2012 e Windows Server 2008 di inviare log a Logs CloudWatch

> **Tip**
>
> CloudWatch include un nuovo agente unificato in grado di raccogliere sia log che metriche dalle istanze EC2 e dai server locali. Ti consigliamo di utilizzare il nuovo agente CloudWatch unificato. Per ulteriori informazioni, consulta Guida introduttiva ai CloudWatch registri.
> Il resto di questa sezione spiega l'uso del vecchio agente Logs. CloudWatch

Consenti alle istanze Amazon EC2 che eseguono Windows Server 2012 e Windows Server 2008 di inviare log a Logs CloudWatch

Utilizza la procedura seguente per consentire alle istanze che eseguono Windows Server 2012 e Windows Server 2008 di inviare i log ai registri. CloudWatch

Download del file di configurazione di esempio

Download il seguente file JSON di esempio su computer: `AWS.EC2.Windows.CloudWatch.json`. Puoi modificarlo seguendo le fasi sotto riportate.

Configurare il file JSON per CloudWatch

Puoi determinare a quali log inviare i dati CloudWatch specificando le tue scelte nel file di configurazione JSON. Il processo di creazione di questo file e l'indicazione delle scelte possono richiedere 30 minuti o più per il completamento. Una volta completata questa attività una volta, puoi riutilizzare il file di configurazione su tutte le istanze.

Fasi

- Passaggio 1: abilitare i log CloudWatch
- Fase 2: Configurare le impostazioni per CloudWatch
- Fase 3: configurazione dei dati da inviare
- Fase 4: configurazione del controllo di flusso

Passaggio 1: abilitare i log CloudWatch

Nella parte superiore del file JSON, modifica "falso" in "vero" per `IsEnabled`:

```
"IsEnabled": true,
```

Fase 2: Configurare le impostazioni per CloudWatch

Specifica le credenziali, la Regione, un nome per il gruppo di log e uno spazio dei nomi del flusso di log. Ciò consente all'istanza di inviare i dati di registro ai CloudWatch registri. Per inviare gli stessi dati di registro a posizioni diverse, puoi aggiungere sezioni aggiuntive con ID univoci (ad esempio, "CloudWatchLogs2" e CloudWatchLogs 3") e una regione diversa per ogni ID.

Per configurare le impostazioni per l'invio dei dati di registro ai registri CloudWatch

1. Nel file JSON, individuare la sezione `CloudWatchLogs`.

   ```
   {
       "Id": "CloudWatchLogs",
       "FullName":
    "AWS.EC2.Windows.CloudWatch.CloudWatchLogsOutput,AWS.EC2.Windows.CloudWatch",
       "Parameters": {
           "AccessKey": "",
           "SecretKey": "",
           "Region": "us-east-1",
           "LogGroup": "Default-Log-Group",
           "LogStream": "{instance_id}"
       }
   },
   ```

2. Lasciare vuoti i campi `AccessKey` e `SecretKey`. Puoi configurare le credenziali tramite un ruolo IAM.
3. In `Region` digitare la Regione alla quale inviare dati di log (ad esempio, `us-east-2`).
4. Per `LogGroup` digita il nome del gruppo di log. Questo nome viene visualizzato nella schermata Log Groups (Gruppi di log) nella console CloudWatch.
5. Per `LogStream` digita il flusso di log di destinazione. Questo nome viene visualizzato nella schermata Log Groups > Streams della CloudWatch console.

   Se si utilizza `{instance_id}`, il nome del flusso di log predefinito è l'ID istanza di tale istanza.

   Se specifichi un nome di log stream che non esiste già, CloudWatch Logs lo crea automaticamente per te. È possibile definire un nome per il flusso di log utilizzando una stringa letterale, le variabili predefinite `{instance_id}`, `{hostname}` e `{ip_address}` oppure una combinazione di queste.

Fase 3: configurazione dei dati da inviare

È possibile inviare i dati del registro degli eventi, i dati di Event Tracing for Windows (ETW) e altri dati di registro ai registri. CloudWatch

Per inviare i dati del registro degli eventi dell'applicazione Windows a Logs CloudWatch

1. Nel file JSON, individuare la sezione `ApplicationEventLog`.

```
{
    "Id": "ApplicationEventLog",
    "FullName":
 "AWS.EC2.Windows.CloudWatch.EventLog.EventLogInputComponent,AWS.EC2.Windows.CloudWatch",
    "Parameters": {
        "LogName": "Application",
        "Levels": "1"
    }
},
```

2. Per `Levels`, specifica il tipo di messaggio da caricare. È possibile specificare uno dei seguenti valori:

   - **1**: vengono caricati solo i messaggi di errore.
   - **2**: vengono caricati solo i messaggi di avviso.
   - **4**: vengono caricati solo i messaggi informativi.

   È possibile combinare i valori per includere diversi tipi di messaggio. Per esempio, il valore **3** carica ad esempio i messaggi di errore (**1**) e i messaggi di avviso (**2**). Il valore **7** carica i messaggi di errore (**1**), i messaggi di avviso (**2**) e i messaggi informativi (**4**).

Per inviare i dati del registro di sicurezza ai CloudWatch registri

1. Nel file JSON, individuare la sezione `SecurityEventLog`.

```
{
    "Id": "SecurityEventLog",
    "FullName":
 "AWS.EC2.Windows.CloudWatch.EventLog.EventLogInputComponent,AWS.EC2.Windows.CloudWatch",
    "Parameters": {
        "LogName": "Security",
```

```
        "Levels": "7"
    }
},
```

2. Per `Levels`, digita **7** per caricare tutti i messaggi.

Per inviare i dati del registro degli eventi di sistema a CloudWatch Logs

1. Nel file JSON, individuare la sezione `SystemEventLog`.

```
{
    "Id": "SystemEventLog",
    "FullName":
 "AWS.EC2.Windows.CloudWatch.EventLog.EventLogInputComponent,AWS.EC2.Windows.CloudWatch",
    "Parameters": {
        "LogName": "System",
        "Levels": "7"
    }
},
```

2. Per `Levels`, specifica il tipo di messaggio da caricare. È possibile specificare uno dei seguenti valori:

   - **1**: vengono caricati solo i messaggi di errore.
   - **2**: vengono caricati solo i messaggi di avviso.
   - **4**: vengono caricati solo i messaggi informativi.

   È possibile combinare i valori per includere diversi tipi di messaggio. Per esempio, il valore **3** carica ad esempio i messaggi di errore (**1**) e i messaggi di avviso (**2**). Il valore **7** carica i messaggi di errore (**1**), i messaggi di avviso (**2**) e i messaggi informativi (**4**).

Per inviare altri tipi di dati del registro degli eventi a CloudWatch Logs

1. Nel file JSON, aggiungere una nuova sezione. Ogni sezione deve avere un `Id` univoco.

```
{
    "Id": "Id-name",
    "FullName":
 "AWS.EC2.Windows.CloudWatch.EventLog.EventLogInputComponent,AWS.EC2.Windows.CloudWatch",
```

```
    "Parameters": {
        "LogName": "Log-name",
        "Levels": "7"
    }
},
```

2. Per `Id` digitare un nome per il log da caricare (ad esempio, **`WindowsBackup`**).
3. Per `LogName`, digita il nome del log da caricare. È possibile trovare il nome del log nel modo seguente.

   a. Aprire il visualizzatore di eventi.

   b. Nel riquadro di navigazione, selezionare Applications and Services Logs (Log di applicazioni e servizi).

   c. Andare al log, quindi scegliere Actions (Operazioni) e Properties (Proprietà).
4. Per `Levels`, specifica il tipo di messaggio da caricare. È possibile specificare uno dei seguenti valori:

   - **1**: vengono caricati solo i messaggi di errore.
   - **2**: vengono caricati solo i messaggi di avviso.
   - **4**: vengono caricati solo i messaggi informativi.

   È possibile combinare i valori per includere diversi tipi di messaggio. Per esempio, il valore **3** carica ad esempio i messaggi di errore (**1**) e i messaggi di avviso (**2**). Il valore **7** carica i messaggi di errore (**1**), i messaggi di avviso (**2**) e i messaggi informativi (**4**).

Per inviare i dati di Event Tracing for Windows ai registri CloudWatch

ETW (traccia eventi per Windows) fornisce meccanismi di log efficienti e dettagliati ai quali le applicazioni possono scrivere log. Ogni ETW è controllato da un gestore di sessione che può avviare e terminare la sessione di log. Ogni sessione dispone di un fornitore e di uno o più consumatori.

1. Nel file JSON, individuare la sezione `ETW`.

```
{
    "Id": "ETW",
    "FullName":
 "AWS.EC2.Windows.CloudWatch.EventLog.EventLogInputComponent,AWS.EC2.Windows.CloudWatch",
    "Parameters": {
```

```
        "LogName": "Microsoft-Windows-WinINet/Analytic",
        "Levels": "7"
    }
},
```

2. Per `LogName`, digita il nome del log da caricare.
3. Per `Levels`, specifica il tipo di messaggio da caricare. È possibile specificare uno dei seguenti valori:

   - **1**: vengono caricati solo i messaggi di errore.
   - **2**: vengono caricati solo i messaggi di avviso.
   - **4**: vengono caricati solo i messaggi informativi.

   È possibile combinare i valori per includere diversi tipi di messaggio. Per esempio, il valore **3** carica ad esempio i messaggi di errore (**1**) e i messaggi di avviso (**2**). Il valore **7** carica i messaggi di errore (**1**), i messaggi di avviso (**2**) e i messaggi informativi (**4**).

Per inviare registri personalizzati (qualsiasi file di registro basato su testo) a Logs CloudWatch

1. Nel file JSON, individuare la sezione `CustomLogs`.

```
{
    "Id": "CustomLogs",
    "FullName":
 "AWS.EC2.Windows.CloudWatch.CustomLog.CustomLogInputComponent,AWS.EC2.Windows.CloudWatch",
    "Parameters": {
        "LogDirectoryPath": "C:\\CustomLogs\\",
        "TimestampFormat": "MM/dd/yyyy HH:mm:ss",
        "Encoding": "UTF-8",
        "Filter": "",
        "CultureName": "en-US",
        "TimeZoneKind": "Local",
        "LineCount": "5"
    }
},
```

2. Per `LogDirectoryPath`, digita il percorso in cui sono memorizzati i log nell'istanza.

3. In `TimestampFormat`, digita il formato timestamp da utilizzare. Per ulteriori informazioni sui valori supportati, consulta l'argomento relativo alle stringhe di formato di data e ora personalizzate su MSDN.

   > **Important**
   >
   > Il file di log di origine deve contenere il timestamp all'inizio di ogni riga di log e uno spazio dopo il timestamp.

4. in `Encoding`, digita la codifica del file da utilizzare (ad esempio, UTF-8). Per ulteriori informazioni sui valori supportati, consulta l'argomento relativo alla classe di codifica su MSDN.

   > **Note**
   >
   > Utilizza il nome di codifica, non il nome di visualizzazione.

5. (Facoltativo) Per `Filter`, digitare il prefisso dei nomi di log. Lascia questo parametro bianco per monitorare tutti i file. Per ulteriori informazioni sui valori supportati, vedere l'argomento FileSystemWatcherFilter Proprietà su MSDN.

6. In `CultureName`, digita le impostazioni locali in cui viene registrato il timestamp (facoltativo). Se `CultureName` è vuoto, si utilizza per impostazione predefinita la stessa configurazione locale utilizzata dall'istanza di Windows. Per ulteriori informazioni relative ai valori supportati, consulta la colonna `Language tag` nella tabella dell'argomento comportamento del prodotto su MSDN.

   > **Note**
   >
   > I valori `div`, `div-MV`, `hu` e `hu-HU` non sono supportati.

7. (Facoltativo) Per `TimeZoneKind`, digitare `Local` o `UTC`. Puoi impostare questo valore per fornire delle informazioni sul fuso orario se non sono incluse nel timestamp del log. Se questo parametro viene lasciato vuoto e se il timestamp non include informazioni sul fuso orario, per impostazione predefinita CloudWatch Logs utilizza il fuso orario locale. Questo parametro viene ignorato se il timestamp contiene già informazioni sul fuso orario.

8. (Facoltativo) Per `LineCount`, digitare il numero di righe dell'intestazione per identificare il file di log. Ad esempio, i file di log IIS dispongono di intestazioni praticamente identiche. È possibile immettere **5** per leggere le prime tre righe dell'intestazione del file di log per identificarlo. Nei file di log IIS, la terza riga è costituita dalla data e dal timestamp, ma non è sempre garantito che il

timestamp sia diverso tra i diversi file di log. Per questo motivo, è consigliabile includere almeno una riga di dati di log reali per identificare in modo univoco il file di log.

Per inviare i dati di registro IIS a Logs CloudWatch

1. Nel file JSON, individuare la sezione `IISLog`.

```
{
    "Id": "IISLogs",
    "FullName":
 "AWS.EC2.Windows.CloudWatch.CustomLog.CustomLogInputComponent,AWS.EC2.Windows.CloudWatch",
    "Parameters": {
        "LogDirectoryPath": "C:\\inetpub\\logs\\LogFiles\\W3SVC1",
        "TimestampFormat": "yyyy-MM-dd HH:mm:ss",
        "Encoding": "UTF-8",
        "Filter": "",
        "CultureName": "en-US",
        "TimeZoneKind": "UTC",
        "LineCount": "5"
    }
},
```

2. Per `LogDirectoryPath`, digita la cartella in cui sono memorizzati i log IIS per un sito individuale (ad esempio, `C:\inetpub\logs\LogFiles\W3SVC`*n*).

> **Note**
>
> È supportato solo il formato di log W3C. I formati IIS, NCSA e quelli personalizzati non sono supportati.

3. In `TimestampFormat`, digita il formato timestamp da utilizzare. Per ulteriori informazioni sui valori supportati, consulta l'argomento relativo alle stringhe di formato di data e ora personalizzate su MSDN.
4. in `Encoding`, digita la codifica del file da utilizzare (ad esempio, UTF-8). Per ulteriori informazioni sui valori supportati, consulta l'argomento relativo alla classe di codifica su MSDN.

> **Note**
>
> Utilizza il nome di codifica, non il nome di visualizzazione.

5. (Facoltativo) Per `Filter`, digitare il prefisso dei nomi di log. Lascia questo parametro bianco per monitorare tutti i file. Per ulteriori informazioni sui valori supportati, vedere l'argomento FileSystemWatcherFilter Proprietà su MSDN.
6. In `CultureName`, digita le impostazioni locali in cui viene registrato il timestamp (facoltativo). Se `CultureName` è vuoto, si utilizza per impostazione predefinita la stessa configurazione locale utilizzata dall'istanza di Windows. Per ulteriori informazioni relative ai valori supportati, consulta la colonna `Language tag` nella tabella dell'argomento comportamento del prodotto su MSDN.

   > **Note**
   >
   > I valori `div`, `div-MV`, `hu` e `hu-HU` non sono supportati.

7. In `TimeZoneKind`, inserisci `Local` o `UTC` (facoltativo). Puoi impostare questo valore per fornire delle informazioni sul fuso orario se non sono incluse nel timestamp del log. Se questo parametro viene lasciato vuoto e se il timestamp non include informazioni sul fuso orario, per impostazione predefinita CloudWatch Logs utilizza il fuso orario locale. Questo parametro viene ignorato se il timestamp contiene già informazioni sul fuso orario.
8. (Facoltativo) Per `LineCount`, digitare il numero di righe dell'intestazione per identificare il file di log. Ad esempio, i file di log IIS dispongono di intestazioni praticamente identiche. È possibile inserire **5** per leggere le prime cinque righe dell'intestazione del file di log per identificarlo. Nei file di log IIS, la terza riga è costituita dalla data e dal timestamp, ma non è sempre garantito che il timestamp sia diverso tra i diversi file di log. Per questo motivo, ti consigliamo di includere almeno una riga di dati di log reali per identificare univocamente il file di log.

Fase 4: configurazione del controllo di flusso

Ogni tipo di dati deve avere una destinazione corrispondente nella sezione `Flows`. Ad esempio, per inviare il registro personalizzato, il registro ETW e il registro di sistema a CloudWatch Logs, aggiungete alla sezione. `(CustomLogs,ETW,SystemEventLog),CloudWatchLogs` `Flows`

> **Warning**
>
> Aggiungere una fase non valida blocca il flusso. Ad esempio, se aggiungi una fase relativa ai parametri del disco, ma la tua istanza non dispone di un disco, tutte le fasi del flusso verranno bloccate.

Puoi inviare lo stesso file di log a più destinazioni. Ad esempio, per inviare il log dell'applicazione a due destinazioni differenti, definite nella sezione `CloudWatchLogs`, aggiungi `ApplicationEventLog,(CloudWatchLogs,CloudWatchLogs2)` alla sezione `Flows`.

Per configurare il controllo di flusso

1. Nel file `AWS.EC2.Windows.CloudWatch.json`, individuare la sezione `Flows`.

```
"Flows": {
    "Flows": [
      "PerformanceCounter,CloudWatch",
      "(PerformanceCounter,PerformanceCounter2), CloudWatch2",
      "(CustomLogs, ETW, SystemEventLog),CloudWatchLogs",
      "CustomLogs, CloudWatchLogs2",
      "ApplicationEventLog,(CloudWatchLogs, CloudWatchLogs2)"
    ]
}
```

2. In `Flows`, aggiungere ciascun tipo di dati da caricare (ad esempio `ApplicationEventLog`) e la sua destinazione (ad esempio `CloudWatchLogs`).

Hai ora terminato la modifica del file JSON. Lo utilizzerai in una fase successiva.

Avvia l'agente

Per consentire a un'istanza Amazon EC2 che esegue Windows Server 2012 o Windows Server 2008 di inviare log a CloudWatch Logs, utilizza il servizio EC2Config (. `EC2Config.exe`) La tua istanza deve avere EC2Config 4.0 o versioni successive; potrai utilizzare questa procedura. Per ulteriori informazioni sull'utilizzo di una versione precedente di EC2Config, consulta Use EC2Config 3.x o precedente per configurare nella CloudWatch Amazon EC2 User Guide for Windows Instances

Per CloudWatch configurare utilizzando EC2Config 4.x

1. Controllare la codifica del file `AWS.EC2.Windows.CloudWatch.json` precedentemente modificato in questa procedura. È supportata solo la codifica UTF-8 senza BOM. Salvare quindi il file nella cartella seguente nell'istanza R2 di Windows Server 2008 - 2012: `C:\Program Files\Amazon\SSM\Plugins\awsCloudWatch\`.
2. Avvia o riavvia l'agente SSM (`AmazonSSMAgent.exe`) utilizzando il pannello di controllo dei servizi di Windows o utilizzando il seguente comando: PowerShell

```
PS C:\> Restart-Service AmazonSSMAgent
```

Dopo il riavvio, l'agente SSM rileva il file di configurazione e configura l'istanza per l'integrazione. CloudWatch Se modifichi i parametri e le impostazioni nel file di configurazione locale, sarà necessario riavviare il SSM Agent per trovare le modifiche. Per disabilitare CloudWatch l'integrazione sull'istanza, apporta `false` e salva `IsEnabled` le modifiche nel file di configurazione.

Avvio rapido: installa l'agente CloudWatch Logs utilizzando AWS OpsWorks and Chef

Puoi installare l'agente CloudWatch Logs e creare flussi di log utilizzando and Chef, uno strumento di automazione dei sistemi AWS OpsWorks e dell'infrastruttura cloud di terze parti. Chef utilizza le "ricette", che puoi scrivere per installare e configurare software sul tuo computer, e i "libri di ricette", che sono raccolte di ricette, per eseguire attività di configurazione e distribuzione delle policy. Per ulteriori informazioni, consulta Chef.

Le ricette di Chef di esempio sotto riportate mostrano come monitorare un file di log su ogni istanza EC2. Le ricette utilizzano il nome dello stack come gruppo di log e il nome host dell'istanza come nome del flusso di log. Per monitorare più file di log, è necessario estendere le ricette per creare più gruppi di log e flussi di log.

Fase 1: creazione di ricette personalizzate

Crea un archivio per archiviare le tue ricette. AWS OpsWorks supporta Git e Subversion oppure puoi archiviare un archivio in Amazon S3. La struttura del repository dei libri di ricette è descritta nella sezione Repository dei libri di ricette nella AWS OpsWorks Guida per l'utente. Gli esempi di seguito presuppongono che il libro di ricette sia denominato `logs`. La ricetta install.rb installa l'agente Logs. CloudWatch Puoi anche scaricare l'esempio del libro di cucina (-Cookbooks.zip). CloudWatchLogs

Crea un file denominato metadata.rb che contiene il seguente codice:

```
#metadata.rb

name             'logs'
version          '0.0.1'
```

Crea il file di configurazione CloudWatch dei registri:

```
#config.rb

template "/tmp/cwlogs.cfg" do
  cookbook "logs"
  source "cwlogs.cfg.erb"
  owner "root"
  group "root"
  mode 0644
end
```

Scarica e installa l'agente CloudWatch Logs:

```
# install.rb

directory "/opt/aws/cloudwatch" do
  recursive true
end

remote_file "/opt/aws/cloudwatch/awslogs-agent-setup.py" do
  source "https://s3.amazonaws.com/aws-cloudwatch/downloads/latest/awslogs-agent-
setup.py"
  mode "0755"
end

  execute "Install CloudWatch Logs agent" do
  command "/opt/aws/cloudwatch/awslogs-agent-setup.py -n -r region -c /tmp/cwlogs.cfg"
  not_if { system "pgrep -f aws-logs-agent-setup" }
end
```

> **Note**
>
> Nell'esempio precedente sostituisci *region (Regione)* con uno dei seguenti: us-east-1, us-west-1, us-west-2, ap-south-1, ap-northeast-2, ap-southeast-1, ap-southeast-2, ap-northeast-1, eu-central-1, eu-west-1, o sa-east-1.

Se l'installazione dell'agente ha esito negativo, verifica che il pacchetto `python-dev` sia installato. In caso contrario, utilizza il comando seguente, quindi riprova a eseguire l'installazione dell'agente:

```
sudo apt-get -y install python-dev
```

Questa ricetta utilizza un file del modello cwlogs.cfg.erb che puoi modificare per specificare diversi attributi, ad esempio quali file registrare. Per ulteriori informazioni su questi attributi, consulta Riferimento dell'agente CloudWatch Logs.

```
[general]
# Path to the AWSLogs agent's state file. Agent uses this file to maintain
# client side state across its executions.
state_file = /var/awslogs/state/agent-state


## Each log file is defined in its own section. The section name doesn't
## matter as long as its unique within this file.
#
#[kern.log]
#
## Path of log file for the agent to monitor and upload.
#
#file = /var/log/kern.log
#
## Name of the destination log group.
#
#log_group_name = kern.log
#
## Name of the destination log stream.
#
#log_stream_name = {instance_id}
#
## Format specifier for timestamp parsing.
#
#datetime_format = %b %d %H:%M:%S
#
#

[<%= node[:opsworks][:stack][:name] %>]
datetime_format = [%Y-%m-%d %H:%M:%S]
log_group_name = <%= node[:opsworks][:stack][:name].gsub(' ','_') %>
file = <%= node[:cwlogs][:logfile] %>
log_stream_name = <%= node[:opsworks][:instance][:hostname] %>
```

Il modello ottiene il nome dello stack e il nome host referenziando gli attributi corrispondenti nella configurazione dello stack e nel JSON di distribuzione. L'attributo che specifica il file da registrare è definito nel file degli attributi default.rb del libro di ricette cwlogs (logs/attributes/default.rb).

```
default[:cwlogs][:logfile] = '/var/log/aws/opsworks/opsworks-agent.statistics.log'
```

Fase 2: Creare uno stack AWS OpsWorks

1. Apri la AWS OpsWorks console all'indirizzo https://console.aws.amazon.com/opsworks/.
2. Nella OpsWorks Dashboard, scegli Aggiungi stack per creare uno AWS OpsWorks stack.
3. Nella schermata Add stack (Aggiungi stack), scegli Chef 11 stack (Stack Chef 11).
4. In Stack name (Nome stack), inserisci un nome.
5. In Use custom Chef Cookbooks (Utilizza i libri di ricette di Chef personalizzati), scegli Yes (Sì).
6. In Repository type (Tipo di repository), seleziona il tipo di repository che utilizzerai. Se utilizzi l'esempio sopra riportato, scegli Http Archive (Archivio Http).
7. In Repository URL (URL del repository), inserisci il repository in cui hai memorizzato il libro di ricette creato nella fase precedente. Se utilizzi l'esempio precedente, immetti **`https://s3.amazonaws.com/aws-cloudwatch/downloads/CloudWatchLogs-Cookbooks.zip`**.
8. Scegli Add Stack (Aggiungi stack) per creare lo stack.

Fase 3: ampliamento del ruolo IAM

Per utilizzare CloudWatch Logs con le tue AWS OpsWorks istanze, devi estendere il ruolo IAM utilizzato dalle tue istanze.

1. Aprire la console IAM all'indirizzo https://console.aws.amazon.com/iam/.
2. Nel riquadro di navigazione scegli Policies (Policy), quindi Create Policy (Crea policy).
3. Nella pagina Create Policy (Crea policy), sotto Create Your Own Policy (Crea la tua policy), scegli Select (Seleziona). Per ulteriori informazioni sulla creazione di policy personalizzate, consulta Policy IAM per Amazon EC2 nella Guida per l'utente di Amazon EC2 per le istanze Linux.
4. Nella pagina Review Policy (Rivedi policy), in Policy Name (Nome policy) digita un nome per la policy.
5. In Policy Document (Documento policy), copia la policy seguente:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
```

```
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents",
        "logs:DescribeLogStreams"
    ],
    "Resource": [
      "arn:aws:logs:*:*:*"
    ]
  }
 ]
}
```

6. Scegliere Create Policy (Crea policy).
7. Nel riquadro di navigazione, scegli Ruoli, quindi nel riquadro dei contenuti, in Nome ruolo, seleziona il nome del ruolo dell'istanza utilizzato dallo stack. AWS OpsWorks Puoi trovare quello utilizzato dal tuo stack nelle impostazioni dello stack (il valore predefinito è `aws-opsworks-ec2-role`).

   > **Note**
   >
   > Seleziona il nome del ruolo, non la casella di controllo.

8. Nella scheda Permissions (Autorizzazioni), in Managed Policies (Policy gestite), seleziona Attach Policy (Collega policy).
9. Nella pagina Attach Policy (Collega policy), nell'intestazione della tabella (accanto a Filter (Filtro) e Search (Cerca)), scegli Policy Type (Tipo di policy), quindi Customer Managed Policies (Policy gestite dal cliente).
10. In Customer Managed Policies (Policy gestite dal cliente), seleziona la policy IAM creata in precedenza e scegli Attach Policy (Collega policy).

    Per ulteriori informazioni sugli utenti e sulle policy, consulta Utenti e gruppi IAM e Gestione delle policy IAM nella Guida per l'utente IAM.

Fase 4: aggiungere un livello

1. Apri la AWS OpsWorks console all'indirizzo https://console.aws.amazon.com/opsworks/.
2. Nel riquadro di navigazione scegli Layers (Livelli).

3. Nel riquadro dei contenuti, seleziona un livello e scegli Add layer (Aggiungi livello).
4. Nella OpsWorksscheda, per Tipo di livello, scegli Personalizzato.
5. Nei campi Name (Nome) e Short Name (Nome breve), inserisci il nome lungo e il nome breve del livello, quindi scegli Add layer (Aggiungi livello).
6. Nella scheda Recipes, in Custom Chef Recipes, ci sono diverse rubriche, Setup, Configure, Deploy, Undeploy e Shutdown, che corrispondono agli eventi del ciclo di vita. AWS OpsWorks AWS OpsWorks attiva questi eventi in questi punti chiave del ciclo di vita dell'istanza, che esegue le ricette associate.

   > **Note**
   >
   > Se le intestazioni sopra riportate non risultano visibili, sotto Custom Chef Recipes (Ricette di Chef personalizzate), scegli edit (modifica).

7. Inserisci logs::config, logs::install accanto a Setup (Installa), scegli + per aggiungerlo all'elenco, quindi scegli Save (Salva).

   AWS OpsWorks esegue questa ricetta su ciascuna delle nuove istanze di questo livello, subito dopo l'avvio dell'istanza.

Fase 5: aggiungere un'istanza

Il livello controlla solo la configurazione delle istanze. Sarà ora necessario aggiungere al livello delle istanze e avviarle.

1. Apri la AWS OpsWorks console all'indirizzo https://console.aws.amazon.com/opsworks/.
2. Nel riquadro di navigazione scegli Instances (Istanze), quindi, sotto il livello, scegli + Instance (+ istanza).
3. Accetta le impostazioni predefinite e scegli Aggiungi istanza per aggiungere l'istanza al livello.
4. Nella colonna Actions (Operazioni) della riga, fai clic su start (avvia) per avviare l'istanza.

   AWS OpsWorks avvia una nuova istanza EC2 e configura CloudWatch i log. Quando è pronto, lo stato dell'istanza diventa online.

Fase 6: visualizzazione dei log

Dovresti vedere il gruppo di log e il flusso di log appena creati nella CloudWatch console dopo che l'agente è stato in esecuzione per alcuni istanti.

Per ulteriori informazioni, consulta Visualizza i dati di registro inviati ai registri CloudWatch .

Segnala lo stato dell'agente CloudWatch Logs

Utilizza la seguente procedura per segnalare lo stato dell'agente CloudWatch Logs sulla tua istanza EC2.

Indicare lo stato dell'agente

1. Connettiti all'istanza EC2. Per ulteriori informazioni, consulta Conessione all'istanza nella Guida per l'utente di Amazon EC2 per le istanze Linux.

   Per ulteriori informazioni sui problemi di connessione, consulta Risoluzione dei problemi di connessione all'istanza nella Guida per l'utente di Amazon EC2 per le istanze Linux
2. Al prompt dei comandi, digita il comando seguente:

   ```
   sudo service awslogs status
   ```

   Se esegui Amazon Linux 2, digita il comando seguente:

   ```
   sudo service awslogsd status
   ```
3. Controlla il file /var/log/awslogs.log per eventuali errori, avvisi o problemi con l' CloudWatch agente Logs.

Avvia l'agente Logs CloudWatch

Se l'agente CloudWatch Logs sulla tua istanza EC2 non si è avviato automaticamente dopo l'installazione, o se hai interrotto l'agente, puoi utilizzare la seguente procedura per avviare l'agente.

Avvio dell'agente

1. Connettiti all'istanza EC2. Per ulteriori informazioni, consulta Conessione all'istanza nella Guida per l'utente di Amazon EC2 per le istanze Linux.

Per ulteriori informazioni sui problemi di connessione, consulta Risoluzione dei problemi di connessione all'istanza nella Guida per l'utente di Amazon EC2 per le istanze Linux.

2. Al prompt dei comandi, digita il comando seguente:

```
sudo service awslogs start
```

Se esegui Amazon Linux 2, digita il comando seguente:

```
sudo service awslogsd start
```

Arresta l'agente Logs CloudWatch

Utilizza la seguente procedura per arrestare l'agente CloudWatch Logs sulla tua istanza EC2.

Arresto dell'agente

1. Connettiti all'istanza EC2. Per ulteriori informazioni, consulta Conessione all'istanza nella Guida per l'utente di Amazon EC2 per le istanze Linux.

   Per ulteriori informazioni sui problemi di connessione, consulta Risoluzione dei problemi di connessione all'istanza nella Guida per l'utente di Amazon EC2 per le istanze Linux.
2. Al prompt dei comandi, digita il comando seguente:

```
sudo service awslogs stop
```

Se esegui Amazon Linux 2, digita il comando seguente:

```
sudo service awslogsd stop
```

Avvio rapido: utilizzalo AWS CloudFormation per iniziare a usare Logs CloudWatch

AWS CloudFormation consente di descrivere e fornire le AWS risorse in formato JSON. I vantaggi di questo metodo includono la possibilità di gestire una raccolta di AWS risorse come singola unità e di replicare facilmente AWS le risorse tra le regioni.

Quando si esegue il provisioning AWS utilizzando AWS CloudFormation, si creano modelli che descrivono le AWS risorse da utilizzare. L'esempio seguente è un frammento di codice di un modello che crea un gruppo di log e un filtro parametri, il quale conta 404 occorrenze e invia questa quantità al gruppo di log.

```
"WebServerLogGroup": {
    "Type": "AWS::Logs::LogGroup",
    "Properties": {
        "RetentionInDays": 7
    }
},

"404MetricFilter": {
    "Type": "AWS::Logs::MetricFilter",
    "Properties": {
        "LogGroupName": {
            "Ref": "WebServerLogGroup"
        },
        "FilterPattern": "[ip, identity, user_id, timestamp, request, status_code =
 404, size, ...]",
        "MetricTransformations": [
            {
                "MetricValue": "1",
                "MetricNamespace": "test/404s",
                "MetricName": "test404Count"
            }
        ]
    }
}
```

Questo è un esempio di base. È possibile configurare distribuzioni di CloudWatch Logs molto più ricche utilizzando. AWS CloudFormationPer ulteriori informazioni sugli esempi di modelli, consulta Amazon CloudWatch Logs Template Snippets nella Guida per l'AWS CloudFormation utente. Per ulteriori informazioni sulle nozioni di base, consulta Nozioni di base di AWS CloudFormation nella Guida per l'utente diAWS CloudFormation .

Utilizzo dei CloudWatch log con un SDK AWS

AWS I kit di sviluppo software (SDK) sono disponibili per molti linguaggi di programmazione più diffusi. Ogni SDK fornisce un'API, esempi di codice, e documentazione che facilitano agli sviluppatori la creazione di applicazioni nel loro linguaggio preferito.

Documentazione sugli SDK	Esempi di codice
AWS SDK for C++	AWS SDK for C++ esempi di codice
AWS SDK for Go	AWS SDK for Go esempi di codice
AWS SDK for Java	AWS SDK for Java esempi di codice
AWS SDK for JavaScript	AWS SDK for JavaScript esempi di codice
SDK AWS for Kotlin	SDK AWS for Kotlin esempi di codice
AWS SDK for .NET	AWS SDK for .NET esempi di codice
AWS SDK for PHP	AWS SDK for PHP esempi di codice
AWS SDK for Python (Boto3)	AWS SDK for Python (Boto3) esempi di codice
AWS SDK for Ruby	AWS SDK for Ruby esempi di codice
AWS SDK for Rust	AWS SDK for Rust esempi di codice
SDK AWS per SAP ABAP	SDK AWS per SAP ABAP esempi di codice
SDK AWS per Swift	SDK AWS per Swift esempi di codice

Per esempi specifici per CloudWatch Logs, vedereEsempi di codice per i CloudWatch log che utilizzano gli SDK AWS.

Esempio di disponibilità

Non riesci a trovare quello che ti serve? Richiedi un esempio di codice utilizzando il link Provide feedback (Fornisci un feedback) nella parte inferiore di questa pagina.

Analisi dei dati di registro con CloudWatch Logs Insights

CloudWatch Logs Insights ti consente di cercare e analizzare in modo interattivo i dati di log in Amazon CloudWatch Logs. Puoi eseguire le query per rispondere in modo rapido ed efficiente a problemi operativi. Se si verifica un problema, puoi utilizzare CloudWatch Logs Insights per identificare potenziali cause e convalidare le correzioni implementate.

CloudWatch Logs Insights include un linguaggio di interrogazione creato appositamente con pochi comandi semplici ma potenti. CloudWatch Logs Insights fornisce query di esempio, descrizioni dei comandi, completamento automatico delle query e individuazione dei campi di registro per aiutarti a iniziare. Query di esempio sono incluse per diversi tipi di log del servizio AWS .

CloudWatch Logs Insights rileva automaticamente i campi nei log di AWS servizi come Amazon Route 53 AWS CloudTraile Amazon VPC e di qualsiasi applicazione o registro personalizzato che emette eventi di registro in formato JSON. AWS Lambda

Puoi utilizzare CloudWatch Logs Insights per cercare i dati di log inviati a CloudWatch Logs il 5 novembre 2018 o successivamente.

Se hai effettuato l'accesso a un account configurato come account di monitoraggio in osservabilità CloudWatch tra account, puoi eseguire query CloudWatch Logs Insights sui gruppi di log negli account di origine collegati a questo account di monitoraggio. È possibile eseguire una query che interroga più gruppi di log situati in account diversi. Per maggiori informazioni, consulta la sezione Osservabilità su più account di CloudWatch .

Una singola richiesta può eseguire query su un massimo di 50 gruppi di log. Le query scadono dopo 60 minuti, se non sono state completate. I risultati delle query sono disponibili per 7 giorni.

Puoi salvare le query create. Ciò ti consente di eseguire query complesse quando necessario, senza doverle ricreare ogni volta che desideri eseguirle.

CloudWatch Le query di Logs Insights comportano addebiti in base alla quantità di dati interrogati. Per ulteriori informazioni, consulta la pagina CloudWatch dei prezzi di Amazon.

> **Important**
>
> Se il tuo team addetto alla sicurezza della rete non consente l'uso di socket Web, al momento non puoi accedere alla parte CloudWatch Logs Insights della CloudWatch console. Puoi

utilizzare le funzionalità di interrogazione di CloudWatch Logs Insights utilizzando le API. Per ulteriori informazioni, StartQueryconsulta Amazon CloudWatch Logs API Reference.

Indice

Guida introduttiva: tutorial delle query

Le sezioni seguenti includono esempi di tutorial sulle query per aiutarti a iniziare a usare Logs Insights. CloudWatch

Argomenti

Tutorial: esecuzione e modifica di una query di esempio

Il seguente tutorial ti aiuta a iniziare a usare Logs Insights. CloudWatch Viene eseguita una query di esempio e descritto come modificarla ed eseguirla nuovamente.

Per eseguire una query, è necessario che i log siano già archiviati in CloudWatch Logs. Se stai già utilizzando CloudWatch Logs e hai configurato gruppi di log e flussi di log, sei pronto per iniziare. Potresti anche avere già dei log se utilizzi servizi come AWS CloudTrailAmazon Route 53 o Amazon

VPC e hai configurato i log di tali servizi in modo che vadano a Logs. CloudWatch Per ulteriori informazioni sull'invio di log a Logs, consulta. CloudWatch Guida introduttiva ai CloudWatch registri

Le query in CloudWatch Logs Insights restituiscono un insieme di campi degli eventi di registro o il risultato di un'aggregazione matematica o di un'altra operazione eseguita sugli eventi di registro. Questo tutorial illustra una query che restituisce un elenco di eventi di log.

Esecuzione di una query di esempio

Per eseguire una query di esempio di Logs CloudWatch Insights

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione scegli Logs (Registri), quindi Logs Insights.

   L'editor di query della pagina Logs Insights contiene una query predefinita che restituisce gli ultimi 20 log eventi.
3. Nell'elenco a discesa Select log group(s) (Seleziona uno o più gruppi di log), scegli uno o più gruppi di log su cui eseguire query.

   Se si tratta di un account di monitoraggio in CloudWatch modalità osservabile su più account, è possibile selezionare gruppi di log negli account di origine e nell'account di monitoraggio. Una singola query può interrogare i log di diversi account contemporaneamente.

   È possibile filtrare i gruppi di log in base al nome del gruppo di log, all'ID account o all'etichetta dell'account.

   Quando si seleziona un gruppo di log, CloudWatch Logs Insights rileva automaticamente i campi di dati nel gruppo. Per visualizzare i campi individuati, seleziona il menu Fields (Campi) in alto a destra nella pagina.
4. (Facoltativo) Utilizza il selettore temporale per selezionare un periodo di tempo per il quale desideri eseguire query.

   Puoi scegliere tra intervalli di 5 e 30 minuti, intervalli di 1, 3 e 12 ore oppure un intervallo di tempo personalizzato.
5. Scegli Run (Esegui) per visualizzare i risultati.

   Per questo tutorial, i risultati includono i 20 log eventi aggiunti più di recente.

CloudWatch Logs visualizza un grafico a barre degli eventi di registro nel gruppo di log nel tempo. Questo grafico a barre mostra la distribuzione di eventi nel gruppo di log che corrisponde alla query e all'intervallo di tempo, non solo gli eventi visualizzati nella tabella.

6. Per visualizzare tutti i campi di un log eventi restituito, scegli l'icona triangolare del menu a discesa a sinistra dell'evento numerato.

Modifica della query di esempio

In questo tutorial, viene modificata la query di esempio per mostrare gli ultimi 50 eventi di log.

Se non hai già eseguito il tutorial precedente, fallo ora. Questo tutorial inizia da dove il tutorial precedente è terminato.

> **Note**
>
> Alcune query di esempio fornite con CloudWatch Logs Insights utilizzano i `tail` comandi `head` or invece di. `limit` Questi comandi sono obsoleti e sostituiti con `limit`. Utilizza `limit` invece di `head` o `tail` in tutte le query di tua creazione.

Per modificare la query di esempio di CloudWatch Logs Insights

1. Nell'editor di query, imposta il valore 20 su 50, e quindi scegli Run (Esegui).

   Vengono visualizzati i risultati della nuova query. Ipotizzando che il gruppo di log contenga dati sufficienti nell'intervallo di tempo predefinito, sono ora elencati 50 eventi di log.

2. (Facoltativo) Puoi salvare le query create. Per salvare questa query, scegli Save (Salva). Per ulteriori informazioni, consulta Salva e riesegui le query di Logs Insights CloudWatch .

Aggiunta di un comando di filtro alla query di esempio

Questo tutorial mostra come apportare una modifica alla query nell'editor di query. In questo tutorial, i risultati della query precedente vengono filtrati in base a un campo negli eventi di log recuperati.

Se non hai già eseguito i tutorial precedenti, fallo in questo momento. Questo tutorial inizia da dove il tutorial precedente è terminato.

Per aggiungere un comando di filtro alla query precedente

1. Stabilisci quale campo filtrare. Per visualizzare i campi più comuni rilevati da CloudWatch Logs negli eventi di registro contenuti nei gruppi di log selezionati negli ultimi 15 minuti e la percentuale di tali eventi di registro in cui viene visualizzato ogni campo, seleziona Campi sul lato destro della pagina.

   Per visualizzare i campi contenuti in un determinato log eventi, scegli l'icona a sinistra di tale riga.

   Il campo awsRegion può essere visualizzato nel log eventi, a seconda degli eventi presenti nei log. Nel resto di questo tutorial, awsRegion viene utilizzato come campo di filtro, ma puoi utilizzarne uno diverso se tale campo non è disponibile.
2. Nella casella Editor di query, posiziona il cursore dopo 50 e premi Invio.
3. Nella nuova riga, digita innanzitutto | (il carattere barra verticale) e uno spazio. I comandi in una query di CloudWatch Logs Insights devono essere separati dal carattere pipe.
4. Specificare **`filter awsRegion="us-east-1"`**.
5. Scegli Esegui.

   La query viene eseguita nuovamente e mostra ora i 50 risultati più recenti che soddisfano il nuovo filtro.

   Se il filtro è stato eseguito su un campo diverso e hai ricevuto un risultato di errore, potrebbe essere necessario applicare un carattere di escape al nome campo. Se il nome campo include caratteri non alfanumerici, è necessario inserire caratteri apice inverso (`) prima e dopo il nome del campo (ad esempio, **`` `error-code`="102" ``**).

   È necessario utilizzare i caratteri di apice inverso per i nomi di campo che contengono caratteri non alfanumerici, ma non per i valori. I valori sono sempre contenuti tra virgolette (").

CloudWatch Logs Insights include potenti funzionalità di interrogazione, tra cui diversi comandi e supporto per espressioni regolari, operazioni matematiche e statistiche. Per ulteriori informazioni, consulta CloudWatch Sintassi delle query di Logs Insights.

Tutorial: esecuzione di una query con una funzione di aggregazione

Puoi utilizzare funzioni di aggregazione con il comando `stats` e come argomenti per altre funzioni. In questo tutorial, viene eseguito un comando di query che conta il numero di log eventi contenenti un

campo specificato. Il comando di query restituisce un conteggio totale raggruppato in base al valore o ai valori del campo specificato. Per ulteriori informazioni sulle funzioni di aggregazione, consulta Operazioni e funzioni supportate nella Amazon CloudWatch Logs User Guide.

Esecuzione di una query con una funzione di aggregazione

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione scegli Logs (Registri), quindi Logs Insights.
3. Nell'elenco a discesa Select log group(s) (Seleziona uno o più gruppi di registro), scegli uno o più gruppi di registro su cui eseguire query.

   Se si tratta di un account di monitoraggio in CloudWatch modalità osservabile su più account, è possibile selezionare gruppi di log negli account di origine e nell'account di monitoraggio. Una singola query può interrogare i log di diversi account contemporaneamente.

   È possibile filtrare i gruppi di log in base al nome del gruppo di log, all'ID account o all'etichetta dell'account.

   Quando si seleziona un gruppo di log, CloudWatch Logs Insights rileva automaticamente i campi di dati nel gruppo. Per visualizzare i campi individuati, seleziona il menu Fields (Campi) in alto a destra nella pagina.
4. Elimina la query predefinita nell'editor di query e immetti il seguente comando:

   ```
   stats count(*) by fieldName
   ```
5. Sostituisci *fieldName* con un campo individuato dal menu Fields (Campi).

   Il menu Campi si trova in alto a destra della pagina e mostra tutti i campi rilevati da CloudWatch Logs Insights nel gruppo di log.
6. Scegli Run (Esegui) per visualizzare i risultati della query.

   I risultati della query mostrano il numero di record nel gruppo di log che corrispondono al comando di query e il conteggio totale raggruppato in base al valore o ai valori del campo specificato.

Esercitazione: eseguire una query che produce una visualizzazione raggruppata per campi di log

Quando si esegue una query che utilizza la funzione `stats` per raggruppare i risultati restituiti in base ai valori di uno o più campi nelle voci del log, è possibile visualizzare i risultati come grafico a barre, grafico a torta, grafico a linee o grafico ad area in pila. In questo modo è possibile visualizzare in modo più efficiente le tendenze nei log.

Per eseguire una query per la visualizzazione

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione scegli Logs (Registri), quindi Logs Insights.
3. Nell'elenco a discesa Select log group(s) (Seleziona uno o più gruppi di registro), scegli uno o più gruppi di registro su cui eseguire query.

   Se si tratta di un account di monitoraggio in CloudWatch modalità osservabile su più account, è possibile selezionare gruppi di log negli account di origine e nell'account di monitoraggio. Una singola query può interrogare i log di diversi account contemporaneamente.

   È possibile filtrare i gruppi di log in base al nome del gruppo di log, all'ID account o all'etichetta dell'account.
4. Nell'editor di query, eliminare i contenuti correnti, immettere la funzione seguente `stats` quanto segue e scegliere Run query (Esegui query).

   ```
   stats count(*) by @logStream
       | limit 100
   ```

   I risultati mostrano il numero di eventi di log nel gruppo di log per ogni flusso di log. I risultati sono limitati a solo 100 righe.
5. Seleziona la scheda Visualization (Visualizzazione).
6. Seleziona la freccia accanto a Line (Linee), quindi scegli Bar (Barre).

   Viene visualizzato il grafico a barre, che mostra una barra per ogni flusso di log nel gruppo di log.

Esercitazione: eseguire una query che produce una visualizzazione serie temporale

Quando esegui una query che utilizza la funzione `bin()` per raggruppare i risultati restituiti da un periodo di tempo, puoi visualizzare i risultati come un grafico a linee, grafico ad area in pila, grafico a torta o grafico a barre. Ciò consente di visualizzare in modo più efficace le tendenze nei log eventi nel tempo.

Per eseguire una query per la visualizzazione

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione scegli Logs (Registri), quindi Logs Insights.
3. Nell'elenco a discesa Select log group(s) (Seleziona uno o più gruppi di registro), scegli uno o più gruppi di registro su cui eseguire query.

   Se si tratta di un account di monitoraggio in CloudWatch modalità osservabile su più account, è possibile selezionare gruppi di log negli account di origine e nell'account di monitoraggio. Una singola query può interrogare i log di diversi account contemporaneamente.

   È possibile filtrare i gruppi di log in base al nome del gruppo di log, all'ID account o all'etichetta dell'account.
4. Nell'editor di query, eliminare i contenuti correnti, immettere la funzione seguente `stats` quanto segue e scegliere Run query (Esegui query).

   ```
   stats count(*) by bin(30s)
   ```

   I risultati mostrano il numero di eventi di registro nel gruppo di log ricevuti da CloudWatch Logs per ogni periodo di 30 secondi.
5. Seleziona la scheda Visualization (Visualizzazione).

   I risultati sono mostrati come un grafico a linee. Per passare a un grafico a barre, grafico a torta o grafico ad area in pila, scegli la freccia accanto a Line (Linea) in alto a sinistra del grafico.

Registri supportati e campi rilevati

CloudWatch Logs Insights supporta diversi tipi di log. Per ogni log inviato ad Amazon CloudWatch Logs, CloudWatch Logs Insights genera automaticamente cinque campi di sistema:

- `@message` contiene il log eventi non analizzato, non elaborato. È l'equivalente del `message` campo in. InputLogevent
- `@timestamp` contiene il timestamp dell'evento nel campo `timestamp` del log eventi. È l'equivalente del `timestamp` campo in InputLogevent.
- `@ingestionTime`contiene l'ora in cui CloudWatch Logs ha ricevuto l'evento di registro.
- `@logStream` contiene il nome del flusso di log a cui il log eventi è stato aggiunto. I flussi di log raggruppano i registri in base allo stesso processo che li ha generati.
- `@log` è un identificatore di gruppo di log sotto forma di *`account-id`*`:`*`log-group-name`*. Nelle query di più gruppi di log, può essere utile per identificare a quale gruppo di log appartiene un particolare evento.

CloudWatch Logs Insights inserisce il simbolo @ all'inizio dei campi generati.

Per molti tipi di log, CloudWatch Logs rileva inoltre automaticamente i campi di log contenuti nei log. Questi campi di rilevamento automatici sono mostrati nella tabella seguente.

Per altri tipi di log con campi che CloudWatch Logs Insights non rileva automaticamente, puoi utilizzare il `parse` comando per estrarre e creare campi estratti da utilizzare in quella query. Per ulteriori informazioni, consulta CloudWatch Sintassi delle query di Logs Insights.

Se il nome di un campo di registro rilevato inizia con il `@` carattere, CloudWatch Logs Insights lo visualizza con un altro `@` aggiunto all'inizio. Ad esempio, se un nome di un campo di log è `@example.com`, questo nome di campo viene visualizzato come `@@example.com`.

Tipo di log	Campi di log rilevati
Log di flusso Amazon VPC	`@timestamp` , `@logStream` , `@message`, `accountId` , `endTime`, `interfaceId` , `logStatus` , `startTime` , `version`, `action`, `bytes`, `dstAddr`, `dstPort`, `packets`, `protocol`, `srcAddr`, `srcPort`
Log di Route 53	`@timestamp` , `@logStream` , `@message`, `edgeLocation` , `ednsClientSubnet` , `hostZoneId` , `protocol`, `queryName` , `queryTimestamp` , `queryType` , `resolverIp` , `responseCode` , `version`
Log di Lambda	`@timestamp` , `@logStream` , `@message`, `@requestId` , `@duration`, `@billedDuration` , `@type`, `@maxMemoryUsed` , `@memorySize`

Tipo di log	Campi di log rilevati
	Se una riga di log Lambda contiene un ID di traccia X-Ray, include anche i seguenti campi: `@xrayTraceId` e `@xraySegmentId` . CloudWatch Logs Insights rileva automaticamente i campi di registro nei log Lambda, ma solo per il primo frammento JSON incorporato in ogni evento di registro. Se un log eventi Lambda contiene più frammenti JSON, puoi analizzare ed estrarre i campi dei log utilizzando il comando **`parse`**. Per ulteriori informazioni, consulta Campi nei registri JSON.
CloudTrail registri Log in formato JSON	Per ulteriori informazioni, consulta Campi nei registri JSON.
Altri tipi di log	`@timestamp` , `@ingestionTime` , `@logStream` , `@message`, `@log`.

Campi nei registri JSON

Con CloudWatch Logs Insights, si utilizza la notazione a punti per rappresentare i campi JSON. Questa sezione contiene un esempio di evento e di frammento di codice JSON che mostra come accedere ai campi JSON utilizzando la notazione con il punto.

Esempio: evento JSON

```
{
    "eventVersion": "1.0",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn: aws: iam: : 123456789012: user/Alice",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "accountId": "123456789012",
        "userName": "Alice"
    },
    "eventTime": "2014-03-06T21: 22: 54Z",
    "eventSource": "ec2.amazonaws.com",
    "eventName": "StartInstances",
```

```
    "awsRegion": "us-east-2",
    "sourceIPAddress": "192.0.2.255",
    "userAgent": "ec2-api-tools1.6.12.2",
    "requestParameters": {
        "instancesSet": {
            "items": [
                {
                    "instanceId": "i-abcde123"
                }
            ]
        }
    },
    "responseElements": {
        "instancesSet": {
            "items": [
                {
                    "instanceId": "i-abcde123",
                    "currentState": {
                        "code": 0,
                        "name": "pending"
                    },
                    "previousState": {
                        "code": 80,
                        "name": "stopped"
                    }
                }
            ]
        }
    }
}
```

L'evento JSON di esempio contiene un oggetto denominato `userIdentity`. `userIdentity` contiene un campo denominato `type`. Per rappresentare il valore di `type` usando la notazione con il punto, utilizza `userIdentity.type`.

L'evento JSON di esempio contiene matrici che si livellano in elenchi di nomi e valori di campi annidati. Per rappresentare il valore di `instanceId` per il primo elemento di `requestParameters.instancesSet`, utilizza `requestParameters.instancesSet.items.0.instanceId`. Il numero `0` posizionato prima del campo `instanceID` si riferisce alla posizione dei valori per il campo `items`. L'esempio seguente contiene un frammento di codice che mostra come è possibile accedere ai campi JSON annidati in un log eventi JSON.

Esempio: query

```
fields @timestamp, @message
| filter requestParameters.instancesSet.items.0.instanceId="i-abcde123"
| sort @timestamp desc
```

Il frammento di codice mostra una query che utilizza la notazione con il punto con il comando `filter` per accedere al valore del campo JSON annidato `instanceId`. La query filtra i messaggi in cui il valore di `instanceId` è uguale a `"i-abcde123"` e restituisce tutti i log eventi che contengono il valore specificato.

> **Note**
>
> CloudWatch Logs Insights può estrarre un massimo di 200 campi di eventi di registro da un registro JSON. Per i campi aggiuntivi che non vengono estratti, è possibile utilizzare il comando `parse` per estrarre questi campi dal log eventi non analizzato e non elaborato nel campo del messaggio. Per ulteriori informazioni sul `parse` comando, consulta la sintassi delle query nella Amazon CloudWatch User Guide.

CloudWatch Sintassi delle query di Logs Insights

Con CloudWatch Logs Insights, si utilizza un linguaggio di query per interrogare i gruppi di log. La sintassi delle query supporta funzioni e operazioni diverse, incluse, a titolo esemplificativo ma non esaustivo, funzioni generali, operazioni aritmetiche e di confronto ed espressioni regolari.

Per creare query che contengono più comandi, separali con la barra verticale (|).

Per creare query che contengono commenti, imposta i commenti con il carattere cancelletto (#).

> **Note**
>
> CloudWatch Logs Insights rileva automaticamente i campi per diversi tipi di log e genera campi che iniziano con il carattere @. Per ulteriori informazioni su questi campi, consulta la sezione Log supportati e campi rilevati nella Amazon CloudWatch User Guide.

La seguente tabella descrive brevemente ogni comando. Di seguito è riportata una descrizione più completa di ogni comando, con esempi.

display	Mostra uno o più campi specifici nei risultati della query.
fields	Mostra campi specifici nei risultati della query e supporta funzioni e operazioni che puoi utilizzare per modificare i valori dei campi e creare nuovi campi da utilizzare nella query.
filter	Filtra la query per restituire solo i log eventi che soddisfano una o più condizioni.
pattern	Raggruppa automaticamente in cluster i dati di log in pattern. Un pattern è una struttura di testo condivisa che ricorre tra i campi dei log.
parse	Estrae i dati da un campo di log per creare un campo estratto che puoi elaborare nella query. **parse** supporta sia la modalità glob con i caratteri jolly che le espressioni regolari.
sort	Mostra i log eventi restituiti in ordine crescente (`asc`) o decrescente (`desc`).
stats	Calcola le statistiche aggregate utilizzando i valori dei campi di log.
limit	Specifica il numero massimo di log eventi che la query deve restituir e. Utile con **sort** per restituire i "primi 20 risultati" o i "20 risultati più recenti".
dedup	Rimuove i risultati duplicati in base a valori specifici nei campi indicati.
unmask	Mostra tutto il contenuto di un log eventi nel quale alcuni contenuti sono mascherati a causa di una policy di protezione dei dati. Per ulteriori informazioni sulla protezione dei dati nei gruppi di log, consulta la sezione Incremento della protezione dei dati di log sensibili con il mascheramento.
Altre operazioni e funzioni	CloudWatch Logs Insights supporta anche molte funzioni e operazion i di confronto, aritmetiche, datetime, numeriche, stringhe, indirizzi IP e funzioni e operazioni generali.

Le sezioni seguenti forniscono maggiori dettagli sui comandi di query di Logs Insights. CloudWatch

Argomenti

display

Utilizza `display` per mostrare uno o più campi specifici nei risultati della query.

Il comando `display` mostra solo i campi specificati. Se la query contiene più comandi `display`, i risultati della query mostrano solo il campo o i campi specificati nel comando `display` finale.

Esempio: visualizzazione di un campo

Il frammento di codice mostra un esempio di query che utilizza il comando parse per estrarre dati da `@message` per creare i campi estratti `loggingType` e `loggingMessage`. La query restituisce i log eventi in cui i valori per `loggingType` sono ERROR. `display` e mostra solo i valori per `loggingMessage` nei risultati della query.

```
fields @message
| parse @message "[*] *" as loggingType, loggingMessage
| filter loggingType = "ERROR"
| display loggingMessage
```

> **Tip**
>
> Utilizzo di `display` solo una volta in una query. Se utilizzi `display` più volte in una query, i risultati della query mostrano i campi specificati nell'ultima occorrenza del comando `display` usato.

campi

Utilizza `fields` per mostrare campi specifici nei risultati della query.

Se la query contiene più comandi `fields` e non include un comando `display`, i risultati mostrano tutti i campi specificati nei comandi `fields`.

Esempio: visualizzazione di campi specifici

Il seguente esempio mostra una query che restituisce 20 log eventi e li visualizza in ordine decrescente. I valori per `@timestamp` e `@message` sono mostrati nei risultati della query.

```
fields @timestamp, @message
| sort @timestamp desc
| limit 20
```

Utilizza `fields` invece di `display`. Quando desideri utilizzare diverse funzioni e operazioni supportate da `fields` per la modifica dei valori dei campi e la creazione di nuovi campi che possono essere utilizzati nelle query.

Puoi utilizzare il comando `fields` con la parola chiave as per creare campi estratti che utilizzano campi e funzioni presenti nel log eventi. Ad esempio, `fields ispresent as isRes` crea un campo estratto denominato `isRes` che può essere utilizzato nel resto della query.

filter

Utilizza `filter` per ottenere log eventi che corrispondono a una o più condizioni.

Esempio: filtrare i log eventi utilizzando una condizione

Il frammento di codice mostra un esempio di query che restituisce tutti i log eventi in cui il valore per `range` è maggiore di 3000. La query limita i risultati a 20 log eventi e li ordina per `@timestamp` e in ordine decrescente.

```
fields @timestamp, @message
| filter (range>3000)
| sort @timestamp desc
| limit 20
```

Esempio: filtrare i log eventi utilizzando più di una condizione

È possibile utilizzare le parole chiave `and` e `or` per combinare più condizioni.

Il frammento di codice mostra un esempio di query che restituisce tutti i log eventi in cui il valore per `range` è maggiore di 3000 e il valore per `accountId` è uguale a 123456789012. La query limita i risultati a 20 log eventi e li ordina per `@timestamp` e in ordine decrescente.

```
fields @timestamp, @message
| filter (range>3000 and accountId=123456789012)
| sort @timestamp desc
| limit 20
```

Corrispondenze ed espressioni regolari nel comando di filtro

Il comando di filtro supporta l'uso di espressioni regolari. Puoi utilizzare i seguenti operatori di confronto (=, `!=`, <, <=, >, >=) e operatori booleani (`and`, `or` e `not`).

Puoi utilizzare la parola chiave `in` per verificare l'appartenenza impostata e la presenza di elementi in una matrice. Per verificare la presenza di elementi in una matrice, posiziona la matrice subito dopo `in`. Puoi utilizzare l'operatore booleano `not` con `in`. Puoi creare query che utilizzano `in` per restituire log eventi in cui i campi sono le stringhe corrispondenti. I campi devono essere stringhe complete. Ad esempio, il seguente frammento di codice mostra una query che utilizza `in` per restituire log eventi in cui il campo `logGroup` è la stringa completa `example_group`.

```
fields @timestamp, @message
| filter logGroup in ["example_group"]
```

Puoi utilizzare le frasi di parole chiave `like` e `not like` per restituire le sottostringhe corrispondenti. Puoi utilizzare l'operatore di espressione regolare =~ per restituire le sottostringhe corrispondenti. Per restituire una sottostringa corrispondente con `like` e `not like`, racchiudi la sottostringa tra virgolette singole o doppie. Puoi utilizzare modelli di espressioni regolari con `like` e `not like`. Per restituire una sottostringa corrispondente con l'operatore dell'espressione regolare,

racchiudi la sottostringa tra le barre. Gli esempi seguenti contengono frammenti di codice che mostrano come è possibile restituire le sottostringhe corrispondenti utilizzando il comando `filter`.

Esempi: corrispondenza di sottostringhe

I tre esempi seguenti restituiscono log eventi in cui `f1` contiene la parola Exception. I tre esempi fanno distinzione tra lettere maiuscole e minuscole.

Il primo esempio restituisce una sottostringa corrispondente con `like`.

```
fields f1, f2, f3
| filter f1 like "Exception"
```

Il secondo esempio restituisce una sottostringa corrispondente con `like` e un modello di espressione regolare.

```
fields f1, f2, f3
| filter f1 like /Exception/
```

Il terzo esempio restituisce una sottostringa corrispondente con un'espressione regolare.

```
fields f1, f2, f3
| filter f1 =~ /Exception/
```

Esempio: corrispondenza di sottostringhe con caratteri jolly

Puoi utilizzare il simbolo dei due punti (`.`) come carattere jolly nelle espressioni regolari per restituire le sottostringhe corrispondenti. Nell'esempio seguente, la query restituisce le corrispondenze in cui il valore per `f1` inizia con la stringa `ServiceLog`.

```
fields f1, f2, f3
| filter f1 like /ServiceLog./
```

Puoi posizionare un asterisco dopo il simbolo del punto (`.*`) per creare un quantificatore greedy che restituisca il maggior numero possibile di corrispondenze. Ad esempio, la query seguente restituisce i risultati in cui il valore per `f1` non inizia solo con la stringa `ServiceLog`, ma include anche la stringa `ServiceLog`.

```
fields f1, f2, f3
```

```
| filter f1 like /ServiceLog.*/
```

Le possibili corrispondenze possono essere formattate come segue:

- `ServiceLogSampleApiLogGroup`
- `SampleApiLogGroupServiceLog`

Esempio: esclusione di sottostringhe dalle corrispondenze

Il seguente esempio mostra una query che restituisce log eventi in cui `f1` non contiene la parola Exception. L'esempio fa distinzione tra maiuscole e minuscole.

```
fields f1, f2, f3
| filter f1 not like "Exception"
```

Esempio: corrispondenza di sottostringhe con modelli che non fanno distinzione tra maiuscole e minuscole

Puoi associare le sottostringhe che non fanno distinzione tra maiuscole e minuscole con espressioni `like` e regolari. Posiziona il parametro (?i) prima della sottostringa corrispondente che desideri restituire. Il seguente esempio mostra una query che restituisce log eventi in cui `f1` contiene la parola Exception o exception.

```
fields f1, f2, f3
| filter f1 like /(?i)Exception/
```

pattern

Utilizza `pattern` per raggruppa automaticamente in cluster i dati di log in pattern.

Un pattern è una struttura di testo condivisa che ricorre tra i campi dei log. È possibile utilizzare `pattern` per evidenziare le tendenze emergenti, monitorare gli errori noti e identificare le righe del log più frequenti o ad alto costo.

Poiché il comando `pattern` identifica automaticamente i pattern più comuni, puoi utilizzarlo come punto di partenza per cercare e analizzare i log. Puoi anche combinare `pattern` con i comandi `filter`, `parse` o `sort` e identificare i pattern nelle query più precise.

Input di comandi nei pattern

Il comando `pattern` prevede uno dei seguenti input: il campo `@message`, un campo estratto creato utilizzando il comando `parse` o una stringa manipolata utilizzando una o più funzioni String.

Output di comandi nei pattern

Il comando `pattern` produce il seguente output:

- `@pattern`: un pattern è una struttura di testo condivisa che ricorre tra i campi dei log eventi. I campi che variano all'interno di un pattern, come l'ID della richiesta o il timestamp, sono rappresentati da `<*>`. Ad esempio, `[INFO] Request time: <*> ms` è un potenziale output per il messaggio di log `[INFO] Request time: 327 ms`.
- `@ratio`: il rapporto tra i log eventi da un periodo di tempo selezionato e i gruppi di log specificati che corrispondono a un pattern identificato. Ad esempio, se metà dei log eventi nei gruppi di log e nel periodo di tempo selezionati corrispondono al pattern, `@ratio` restituisce `0.50`
- `@sampleCount`: il numero di log eventi da un periodo di tempo selezionato e i gruppi di log specificati che corrispondono a un pattern identificato.
- `@severityLabel`: la gravità o il livello del log, che indica il tipo di informazioni contenute in un log. Ad esempio, `Error`, `Warning`, `Info` o `Debug`.

Examples (Esempi)

Il comando seguente identifica i log con strutture simili in gruppi di log specificati nell'intervallo di tempo selezionato, raggruppandoli per pattern e numero

```
pattern @message
```

Il comando `pattern` può essere usato in combinazione con il comando `filter`

```
filter @message like /ERROR/
| pattern @message
```

Il comando `pattern` può essere utilizzato con i comandi `parse` e `sort`

```
filter @message like /ERROR/
| parse @message 'Failed to do: *' as cause
| pattern cause
| sort @sampleCount asc
```

parse

Utilizza `parse` per estrarre i dati da un campo di log e creare un campo estratto che puoi elaborare nella query. **`parse`** supporta sia la modalità glob con i caratteri jolly che le espressioni regolari.

Puoi analizzare i campi JSON annidati con un'espressione regolare.

Esempio: analisi di un campo JSON annidato

Il frammento di codice mostra come analizzare un log eventi JSON che è stato dissociato durante l'acquisizione.

```
{'fieldsA': 'logs', 'fieldsB': [{'fA': 'a1'}, {'fA': 'a2'}]}
```

Il frammento di codice mostra una query con un'espressione regolare che estrae i valori per `fieldsA` e `fieldsB` per creare i campi estratti `fld` e `array`.

```
parse @message "'fieldsA': '*', 'fieldsB': ['*']" as fld, array
```

Gruppi di acquisizione denominati

Quando si utilizza **`parse`** con un'espressione regolare, è possibile utilizzare gruppi di acquisizione denominati per acquisire un pattern in un campo. La sintassi è `parse @message (?<Name>pattern)`.

L'esempio seguente utilizza un gruppo di acquisizione su un log di flusso VPC per estrarre l'ENI in un campo denominato `NetworkInterface`.

```
parse @message /(?<NetworkInterface>eni-.*?) / display @timestamp, NetworkInterface
```

> **Note**
>
> I log eventi JSON vengono appiattiti durante l'acquisizione. Attualmente, l'analisi dei campi JSON annidati con un'espressione globale non è supportata. È possibile analizzare solo i log eventi JSON che includono non più di 200 campi di log eventi. Quando si analizzano i campi JSON annidati, è necessario formattare l'espressione regolare nella query in modo che corrisponda al formato del log eventi JSON.

Esempi del comando parse

Utilizza un'espressione glob per estrarre i campi **`@user`**, **`@method`** e **`@latency`** dal campo di log **`@message`** e restituire la latenza media per ogni combinazione univoca di **`@method`** e **`@user`**.

```
parse @message "user=*, method:*, latency := *" as @user,
    @method, @latency | stats avg(@latency) by @method,
    @user
```

Utilizza un'espressione regolare per estrarre i campi temporanei **`@user2`**, **`@method2`** e **`@latency2`** dal campo di log **`@message`** e restituire la latenza media per ogni combinazione univoca di **`@method2`** e **`@user2`**.

```
parse @message /user=(?<user2>.*?), method:(?<method2>.*?),
    latency := (?<latency2>.*?)/ | stats avg(latency2) by @method2,
    @user2
```

Estrae i campi **`loggingTime`**, **`loggingType`** e **`loggingMessage`**, filtra per log eventi che contengono le stringhe **`ERROR`** o **`INFO`** e quindi mostra solo i campi **`loggingMessage`** e **`loggingType`** per gli eventi che contengono una stringa **`ERROR`**.

```
FIELDS @message
    | PARSE @message "* [*] *" as loggingTime, loggingType, loggingMessage
    | FILTER loggingType IN ["ERROR", "INFO"]
    | DISPLAY loggingMessage, loggingType = "ERROR" as isError
```

sort

Utilizza `sort` per visualizzare i log eventi in ordine crescente (`asc`) o decrescente (`desc`) di un campo specificato. Puoi utilizzare questa opzione con il comando `limit` per creare query "primi N" o "ultimi N".

Ad esempio, la seguente query per i log di flusso di Amazon VPC individua i primi 15 trasferimenti di pacchetti tra gli host.

```
stats sum(packets) as packetsTransferred by srcAddr, dstAddr
    | sort packetsTransferred  desc
    | limit 15
```

statistiche

Utilizza `stats` per creare visualizzazioni dei dati di log come grafici a barre, grafici a linee e grafici ad area in pila. Ciò consente di identificare in modo più efficiente i modelli nei dati di registro. CloudWatch Logs Insights genera visualizzazioni per le query che utilizzano la `stats` funzione e una o più funzioni di aggregazione.

Ad esempio, la seguente query in un gruppo di log Route 53 restituisce visualizzazioni che mostrano la distribuzione dei record di Route 53 all'ora, per tipo di query.

```
stats count(*) by queryType, bin(1h)
```

Tutte queste query possono produrre grafici a barre. Se la query utilizza la funzione `bin()` per raggruppare i dati per un campo nel tempo, è anche possibile visualizzare grafici a linee e grafici ad area in pila.

Argomenti

- Visualizzazione dei dati di serie temporali
- Visualizzazione dei dati di log raggruppati per campi
- Utilizzo di più comandi stats in un'unica query
- Funzioni da utilizzare con le statistiche

Visualizzazione dei dati di serie temporali

Le visualizzazioni delle serie temporali funzionano per le query con le seguenti caratteristiche:

- La query contiene una o più funzioni di aggregazione. Per ulteriori informazioni, consulta Aggregation Functions in the Stats Command.
- La query utilizza la funzione `bin()` per raggruppare i dati di un campo.

Queste query possono produrre grafici a linee, grafici ad area in pila, grafici a barre e grafici a torta.

Examples (Esempi)

Per un tutorial completo, consulta the section called “Esercitazione: eseguire una query che produce una visualizzazione serie temporale”.

Di seguito sono riportati altri esempi di query che funzionano per la visualizzazione delle serie temporali.

La seguente query genera una visualizzazione dei valori medi del campo `myfield1`, con un punto dati creato ogni cinque minuti. Ogni punto dati è l'aggregazione delle medie dei valori `myfield1` dei log dei cinque minuti precedenti.

```
stats avg(myfield1) by bin(5m)
```

La seguente query genera una visualizzazione di tre valori basati su campi diversi, con un punto dati creato ogni cinque minuti. La visualizzazione viene generata perché la query contiene funzioni di aggregazione e usa `bin()` come campo di raggruppamento.

```
stats avg(myfield1), min(myfield2), max(myfield3) by bin(5m)
```

Restrizioni del grafico a linee e del grafico ad area in pila

Le query che aggregano le informazioni sulle voci di log, ma non utilizzano la funzione `bin()`, possono generare grafici a barre. Tuttavia, le query non possono generare grafici a linee o grafici ad area in pila. Per ulteriori informazioni su questi tipi di query, consulta the section called “Visualizzazione dei dati di log raggruppati per campi”.

Visualizzazione dei dati di log raggruppati per campi

È possibile produrre grafici a barre per le query che utilizzano la funzione `stats` e una o più funzioni di aggregazione. Per ulteriori informazioni, consulta Aggregation Functions in the Stats Command.

Per visualizzare la visualizzazione, eseguire la query. Quindi scegliere la scheda Visualization (Visualizzazione) selezionare la freccia accanto a Linea (Linea), e scegliere Bar (barra). Le visualizzazioni sono limitate a un massimo di 100 barre nel grafico a barre.

Examples (Esempi)

Per un tutorial completo, consulta the section called “Esercitazione: eseguire una query che produce una visualizzazione raggruppata per campi di log”. I paragrafi seguenti includono ulteriori query di esempio per la visualizzazione in base ai campi.

La seguente query di log di flusso VPC trova il numero medio di byte trasferiti per sessione per ogni indirizzo di destinazione.

```
stats avg(bytes) by dstAddr
```

È inoltre possibile produrre un grafico che include più di una barra per ogni valore risultante. Ad esempio, la query del log di flusso VPC seguente trova il numero medio e massimo di byte trasferiti per sessione per ogni indirizzo di destinazione.

```
stats avg(bytes), max(bytes) by dstAddr
```

La query seguente trova il numero di registro di query Amazon Route 53 per ogni tipo di query.

```
stats count(*) by queryType
```

Utilizzo di più comandi stats in un'unica query

È possibile utilizzare fino a due comandi `stats` in un'unica query. Ciò consente di eseguire un'aggregazione aggiuntiva sull'output della prima aggregazione.

Esempio: interrogazione con due comandi **`stats`**

Ad esempio, la seguente query trova innanzitutto il volume di traffico totale in contenitori da 5 minuti, quindi calcola il volume di traffico più alto, più basso e medio tra questi contenitori da 5 minuti.

```
FIELDS strlen(@message) AS message_length
| STATS sum(message_length)/1024/1024 as logs_mb BY bin(5m)
| STATS max(logs_mb) AS peak_ingest_mb,
        min(logs_mb) AS min_ingest_mb,
        avg(logs_mb) AS avg_ingest_mb
```

Esempio: combinazione di più comandi stats con altre funzioni come **`filter`**, **`fields`** e **`bin`**

È possibile combinare due comandi `stats` con altri comandi come `filter` e `fields` in un'unica query. Ad esempio, la seguente query trova il numero di indirizzi IP distinti nelle sessioni e trova il numero di sessioni per piattaforma client, filtra tali indirizzi IP e infine trova la media delle richieste di sessione per piattaforma client.

```
STATS count_distinct(client_ip) AS session_ips,
      count(*) AS requests BY session_id, client_platform
| FILTER session_ips > 1
```

```
| STATS count(*) AS multiple_ip_sessions,
        sum(requests) / count(*) AS avg_session_requests BY client_platform
```

È possibile utilizzare le funzioni `bin` e `dateceil` nelle query con più comandi `stats`. Ad esempio, la seguente query combina prima i messaggi in blocchi da 5 minuti, quindi aggrega i blocchi da 5 minuti in blocchi da 10 minuti e calcola i volumi di traffico più alti, più bassi e medi all'interno di ogni blocco di 10 minuti.

```
FIELDS strlen(@message) AS message_length
| STATS sum(message_length) / 1024 / 1024 AS logs_mb BY BIN(5m) as @t
| STATS max(logs_mb) AS peak_ingest_mb,
        min(logs_mb) AS min_ingest_mb,
        avg(logs_mb) AS avg_ingest_mb BY dateceil(@t, 10m)
```

Note e limitazioni

Una query può avere al massimo due comandi `stats`. Questa quota non può essere modificata.

Se si utilizza un comando `sort` o `limit`, questo deve apparire dopo il secondo comando `stats`. Se è precedente al secondo comando `stats`, la query non è valida.

Quando una query ha due comandi `stats`, i risultati parziali della query non iniziano a essere visualizzati fino al completamento della prima aggregazione `stats`.

Nel secondo comando `stats` di un'unica query, è possibile fare riferimento solo ai campi definiti nel primo comando `stats`. Ad esempio, la seguente query non è valida perché il campo `@message` non sarà disponibile dopo la prima aggregazione `stats`.

```
FIELDS @message
| STATS SUM(Fault) by Operation
# You can only reference `SUM(Fault)` or Operation at this point
| STATS MAX(strlen(@message)) AS MaxMessageSize # Invalid reference to @message
```

Tutti i campi a cui si fa riferimento dopo il primo comando `stats` devono essere definiti in tale primo comando `stats`.

```
STATS sum(x) as sum_x by y, z
| STATS max(sum_x) as max_x by z
# You can only reference `max(sum_x)`, max_x or z at this point
```

Important

La funzione `bin` utilizza sempre implicitamente il campo `@timestamp`. Ciò significa che non è possibile utilizzare `bin` nel secondo comando `stats` senza utilizzare il primo comando `stats` per propagare il campo `timestamp`. Ad esempio, la query seguente non è valida.

```
FIELDS strlen(@message) AS message_length
 | STATS sum(message_length) AS ingested_bytes BY @logStream
 | STATS avg(ingested_bytes) BY bin(5m) # Invalid reference to @timestamp field
```

Definisci invece il campo `@timestamp` nel primo comando `stats`, quindi potrai utilizzarlo con `dateceil` nel secondo comando `stats`, come nell'esempio seguente.

```
FIELDS strlen(@message) AS message_length
 | STATS sum(message_length) AS ingested_bytes, max(@timestamp) as @t BY
 @logStream
 | STATS avg(ingested_bytes) BY dateceil(@t, 5m)
```

Funzioni da utilizzare con le statistiche

CloudWatch Logs Insights supporta sia le funzioni di aggregazione delle statistiche che le funzioni non di aggregazione delle statistiche.

Utilizza funzioni di aggregazione delle statistiche nel comando `stats` e come argomenti per altre funzioni.

Funzione	Tipo di risultato	Descrizione
`avg(fieldName: NumericLogField)`	number	La media dei valori nel campo specificato.

`count()`
 	
`count(fieldName: LogField)`
	number	Conta i log eventi. `count()` (o `count(*)`) conta tutti gli eventi restituiti dalla query, mentre `count(fieldName)` conta tutti i registri che includono il nome di campo specificato.

Funzione	Tipo di risultato	Descrizione
`count_distinct(fieldName: LogField)`	number	Restituisce il numero di valori univoci per il campo. Se il campo dispone di alta cardinalità (contiene molti valori univoci), il valore restituito da `count_distinct` è solo un'approssimazione.
`max(fieldName: LogField)`	LogFieldValue	Il numero massimo di valori per questo campo di log nei log di query.
`min(fieldName: LogField)`	LogFieldValue	Il numero minimo di valori per questo campo di log nei log di query.
`pct(fieldName: LogFieldValue, percent: number)`	LogFieldValue	Un percentile indica lo stato relativo di un valore in un set di dati. Ad esempio, `pct(@duration, 95)` restituisce il valore `@duration` a cui il 95% dei valori di `@duration` sono inferiori a questo valore e il 5% sono superiori a questo valore.
`stddev(fieldName: NumericLogField)`	number	La deviazione standard dei valori nel campo specificato.
`sum(fieldName: NumericLogField)`	number	La somma dei valori nel campo specificato.

Funzioni di non aggregazione delle statistiche

Utilizza funzioni di non aggregazione nel comando `stats` e come argomenti per altre funzioni.

Funzione	Tipo di risultato	Descrizione
`earliest(fieldName: LogField)`	LogField	Restituisce il valore di `fieldName` dal log eventi con il timestamp meno recente nei log oggetto della query.
`latest(fieldName: LogField)`	LogField	Restituisce il valore di `fieldName` dal log eventi con il timestamp più recente nei log oggetto della query.
`sortsFirst(fieldName: LogField)`	LogField	Restituisce il valore di `fieldName` che occupa la prima posizione dell'ordine nei log oggetto della query.
`sortsLast(fieldName: LogField)`	LogField	Restituisce il valore di `fieldName` che occupa l'ultima posizione dell'ordine nei log oggetto della query.

limit

Utilizza `limit` per specificare il numero di log eventi che la query deve restituire.

L'esempio seguente restituisce solo i 25 log eventi più recenti

```
fields @timestamp, @message | sort @timestamp desc | limit 25
```

dedup

Utilizza `dedup` per rimuovere i risultati duplicati in base a valori specifici nei campi indicati. Puoi utilizzare `dedup` con uno o più campi. Se specifichi un campo con `dedup`, viene restituito un solo log eventi per ogni valore univoco di tale campo. Se specifichi più campi, viene restituito un log eventi per ogni combinazione univoca di valori per tali campi.

I duplicati vengono eliminati in base al criterio di ordinamento, conservando solo il primo risultato dell'ordinamento. Ti consigliamo di ordinare i risultati prima di sottoporli al comando `dedup`. Se i risultati non vengono ordinati prima di essere sottoposti a `dedup`, viene utilizzato l'ordinamento decrescente predefinito tramite `@timestamp`.

I valori nulli non sono considerati duplicati per la valutazione. I log eventi con valori nulli per uno qualsiasi dei campi specificati vengono conservati. Per eliminare i campi con valori nulli, utilizza **`filter`** tramite la funzione `isPresent(field)`.

L'unico comando di query che puoi utilizzare in una query dopo il comando `dedup` è `limit`.

Esempio: visualizza solo il log eventi più recente per ogni valore univoco del campo denominato **`server`**

L'esempio seguente mostra i campi `timestamp`, `server`, `severity` e `message` solo per l'evento più recente per ogni valore univoco di `server`.

```
fields @timestamp, server, severity, message
| sort @timestamp asc
| dedup server
```

Per altri esempi di query di CloudWatch Logs Insights, consulta. Query generali

unmask

Utilizza `unmask` per visualizzare tutto il contenuto di un log eventi nel quale alcuni contenuti sono mascherati a causa di una policy di protezione dei dati. Per eseguire questo comando, è necessario disporre dell'autorizzazione `logs:Unmask`.

Per ulteriori informazioni sulla protezione dei dati nei gruppi di log, consulta la sezione Incremento della protezione dei dati di log sensibili con il mascheramento.

Funzioni booleane, di confronto, numeriche, datetime e altre

CloudWatch Logs Insights supporta molte altre operazioni e funzioni nelle query, come spiegato nelle sezioni seguenti.

Argomenti

- Operatori aritmetici
- Operatori booleani
- Operatori di confronto
- Operatori numerici
- Funzioni DateTime

- Funzioni generali
- Funzioni della stringa di indirizzi IP
- Funzioni stringa

Operatori aritmetici

Le operazioni aritmetiche accettano tipi di dati numerici come argomenti e restituiscono risultati numerici. Puoi utilizzare operazioni aritmetiche nei comandi `filter` e `fields` e come argomenti per altre funzioni.

Operazione	Descrizione
`a + b`	Addizione
`a - b`	Sottrazione
`a * b`	Moltiplicazione
`a / b`	Divisione
`a ^ b`	Elevamento a potenza (`2 ^ 3` restituisce 8)
`a % b`	Resto o modulo (`10 % 3` restituisce 1)

Operatori booleani

Utilizza gli operatori booleani **`and`**, **`or`** e **`not`**.

> **Note**
>
> Utilizza gli operatori booleani solo nelle funzioni che restituiscono un valore TRUE o FALSE.

Operatori di confronto

Le operazioni di confronto accettano tutti i tipi di dati come argomenti e restituiscono un risultato booleano. Utilizza operazioni di confronto nel comando `filter` e come argomenti per altre funzioni.

Operatore	Descrizione
=	Uguale
!=	Non uguale
<	Minore di
>	Maggiore di
<=	Minore o uguale a
>=	Maggiore o uguale a

Operatori numerici

Le operazioni numeriche accettano tipi di dati numerici come argomenti e restituiscono risultati numerici. Utilizza operazioni aritmetiche nei comandi `filter` e `fields`, oltre che come argomenti per altre funzioni.

Operazione	Tipo di risultato	Descrizione
`abs(a: number)`	number	Valore assoluto
`ceil(a: number)`	number	Arrotonda per eccesso (l'intero più piccolo che è maggiore del valore di a)
`floor(a: number)`	number	Arrotonda per difetto (l'intero più grande che è minore del valore di a)
`greatest(a: number, ...numbers: number[])`	number	Restituisce il valore più grande

Operazione	Tipo di risultato	Descrizione
`least(a: number, ...numbers: number[])`	number	Restituisce il valore di piccolo
`log(a: number)`	number	Log naturale
`sqrt(a: number)`	number	Radice quadrata

Funzioni DateTime

Funzioni DateTime

Utilizza le funzioni datetime nei comandi `fields` e `filter` e come argomenti per altre funzioni. Utilizza queste funzioni per creare bucket temporali per le query con funzioni di aggregazione. Puoi utilizzare anche periodi di tempo costituiti da un numero e quindi `m` per i minuti o `h` per le ore. Ad esempio, `10m` è 10 minuti e `1h` è un'ora. La tabella seguente contiene un elenco delle diverse funzioni datetime che è possibile utilizzare nei comandi di query. La tabella elenca il tipo di risultato di ciascuna funzione e contiene una descrizione di ciascuna funzione.

> **Tip**
>
> Quando crei un comando di query, puoi utilizzare il selettore temporale per selezionare un periodo di tempo per il quale desideri eseguire query. Ad esempio, puoi impostare un periodo di tempo con intervalli di 5 e 30 minuti, intervalli di 1, 3 e 12 ore oppure un intervallo di tempo personalizzato. Puoi anche impostare periodi di tempo tra date specifiche.

Funzione	Tipo di risultato	Descrizione
`bin(period: Period)`	Timestamp	Arrotonda il valore di `@timestamp` al periodo di tempo specificato per poi troncarlo . Ad esempio, `bin(5m)` arrotonda il valore `@timestamp` ai 5 minuti più vicini.

Funzione	Tipo di risultato	Descrizione
		È possibile utilizzarlo per raggruppare più voci di log in un'interrogazione. L'esempio seguente restituisce il numero di eccezioni all'ora: `filter @message like /Exception/	stats count(*) as exceptionCount by bin(1h)	sort exceptionCount desc` La funzione `bin` supporta le seguenti abbreviazioni e unità di tempo. Per tutte le unità e le abbreviazioni che includono più di un carattere, è supportata l'aggiunta di `s` per il plurale. Quindi entrambi `hr` e `hrs` lavorano per specificare gli orari. • `millisecond ms msec` • `second s sec` • `minute m min` • `hour h hr` • `day d` • `week w` • `month mo mon` • `quarter q qtr` • `year y yr`
`datefloor(timestamp: Timestamp, period: Period)`	Timestamp	Tronca il time stamp al periodo specificato. Ad esempio, `datefloor(@timestamp, 1h)` tronca tutti i valori di `@timestamp` alla mezzora.

Funzione	Tipo di risultato	Descrizione
`dateceil(timestamp : Timestamp, period: Period)`	Timestamp	Arrotonda il time stamp al periodo specifica to e quindi tronca. Ad esempio, `dateceil( @timestamp, 1h)` tronca tutti i valori di `@timestamp` all'inizio di ogni ora.
`fromMillis(fieldNa me: number)`	Timestamp	Interpreta il campo di input come il numero di millisecondi dall'epoca Unix e lo converte in un timestamp.
`toMillis(fieldName: Timestamp)`	number	Converte il timestamp trovato nel campo denominato in un numero che rappresenta i millisecondi dall'epoca di Unix. Ad esempio, `toMillis(@timestamp)` converte il timestamp `2022-01-14T13:18:0 31.000-08:00` in `1642195111000` .

> **Note**
>
> Attualmente, CloudWatch Logs Insights non supporta il filtraggio dei log con timestamp leggibili dall'uomo.

Funzioni generali

Funzioni generali

Utilizza le funzioni generali nei comandi `fields` e `filter` e come argomenti per altre funzioni.

Funzione	Tipo di risultato	Descrizione
`ispresent(fieldName: LogField)`	Booleano	Restituisce `true` se il campo esiste

Funzione	Tipo di risultato	Descrizione
`coalesce(fieldName: LogField, ...fieldNames: LogField[])`	LogField	Restituisce il primo valore non nullo dall'elenco.

Funzioni della stringa di indirizzi IP

Funzioni della stringa di indirizzi IP

Utilizza le funzioni stringa per gli indirizzi IP nei comandi `filter` e `fields`, oltre che come argomenti per altre funzioni.

Funzione	Tipo di risultato	Descrizione
`isValidIp(fieldName: string)`	booleano	Restituisce `true` se il campo è un indirizzo IPv4 o IPv6 valido.
`isValidIpV4(fieldName: string)`	booleano	Restituisce `true` se il campo è un indirizzo IPv4 valido.
`isValidIpV6(fieldName: string)`	booleano	Restituisce `true` se il campo è un indirizzo IPv6 valido.
`isIpInSubnet(fieldName: string, subnet: string)`	booleano	Restituisce `true` se il campo è un indirizzo IPv4 o IPv6 valido all'interno della sottorete v4 o v6 specificata. Quando si specifica la sottorete, utilizzare la notazione CIDR, come `192.0.2.0/24` o `2001:db8::/32` , dove `192.0.2.0` o `2001:db8::` è l'inizio del blocco CIDR.
`isIpv4InSubnet(fieldName: string, subnet: string)`	booleano	Restituisce `true` se il campo è un indirizzo IPv4 valido all'interno della sottorete v4 specificata. Quando si specifica la sottorete, utilizzare la notazione CIDR, come `192.0.2.0`

Funzione	Tipo di risultato	Descrizione
		`/24` , dove `192.0.2.0` è l'inizio del blocco CIDR.
`isIpv6InSubnet(fieldName: string, subnet: string)`	booleano	Restituisce `true` se il campo è un indirizzo IPv6 valido all'interno della sottorete v6 specificata. Quando si specifica la sottorete, utilizzare la notazione CIDR, come `2001:db8::/32` , dove `2001:db8::` è l'inizio del blocco CIDR.

Funzioni stringa

Funzioni stringa

Utilizza le funzioni stringa nei comandi `fields` e `filter`, oltre che come argomenti per altre funzioni.

Funzione	Tipo di risultato	Descrizione
`isempty(fieldName: string)`	Numero	Restituisce 1 se il campo manca o è una stringa vuota.
`isblank(fieldName: string)`	Numero	Restituisce 1 se il campo manca, è una stringa vuota o contiene solo spazi vuoti.
`concat(str: string, ...strings: string[])`	string	Concatena le stringhe.

`ltrim(str: string)`
 	
`ltrim(str: string, trimChars: string)`
	string	Se la funzione non dispone di un secondo argomento, rimuove gli spazi vuoti dalla sinistra della stringa. Se la funzione dispone di un

Funzione	Tipo di risultato	Descrizione
		secondo argomento stringa, non rimuove gli spazi vuoti. Invece, rimuove i caratteri in `trimChars` dalla sinistra di `str`. Ad esempio, `ltrim("xyZxyfooxyZ","xyZ")` restituisce `"fooxyZ"`.

`rtrim(str: string)`
 	
`rtrim(str: string, trimChars: string)`
	string	Se la funzione dispone di un secondo argomento stringa, rimuove gli spazi vuoti dalla destra della stringa. Se la funzione dispone di un secondo argomento stringa, non rimuove gli spazi vuoti. Invece, rimuove i caratteri di `trimChars` dalla destra di `str`. Ad esempio, `rtrim("xyZfooxyxyZ","xyZ")` restituisce `"xyZfoo"`.

`trim(str: string)`
 	
`trim(str: string, trimChars: string)`
	string	Se la funzione dispone di un secondo argomento, rimuove gli spazi vuoti da entrambe le estremità della stringa. Se la funzione dispone di un secondo argomento stringa, non rimuove gli spazi vuoti. Invece, rimuove i caratteri di `trimChars` da entrambi i lati di `str`. Ad esempio, `trim("xyZxyfooxyxyZ","xyZ")` restituisce `"foo"`.

Funzione	Tipo di risultato	Descrizione
`strlen(str: string)`	number	Restituisce la lunghezza della stringa in punti di codice Unicode.
`toupper(str: string)`	string	Converte la stringa in maiuscolo.
`tolower(str: string)`	string	Converte la stringa in minuscolo.

`substr(str: string, startIndex: number)`
 	
`substr(str: string, startIndex: number, length: number)`
	string	Restituisce una sottostringa partendo dall'indice specifica to dall'argomento numero fino alla fine della stringa. Se la funzione dispone di un secondo argomento numero, contiene la lunghezza della sottostringa da recuperare. Ad esempio, `substr("x yZfooxyZ",3, 3)` restituisce `"foo"`.

`replace(fieldName: string, searchValue: string, replaceVa lue: string)`	string	Sostituisce tutte le istanze di `searchValue` in `fieldName: string` con `replaceValue` . Ad esempio, la funzione `replace(logGroup," smoke_test","Smoke ")` cerca log eventi in cui il campo `logGroup` contiene il valore della stringa `smoke_test` e sostituisce il valore con la stringa `Smoke`.

Funzione	Tipo di risultato	Descrizione
`strcontains(str: string, searchValue: string)`	number	Restituisce 1 se `str` contiene `searchValue` e 0 in caso contrario.

Campi che contengono caratteri speciali

Tutti i campi di log nominati nelle query e che includono caratteri diversi dal simbolo `@`, dal punto (`.`) e dai caratteri non alfanumerici devono essere racchiusi tra caratteri apice inverso (`` ` ``). Ad esempio, il campo di log `foo-bar` deve essere racchiuso tra caratteri apice inverso (`` `foo-bar` ``) perché contiene un carattere non alfanumerico, ossia il trattino (`-`).

Utilizzo di alias e commenti nelle query

Crea query che contengono alias. Utilizza gli alias per rinominare i campi di log o per estrarre valori nei campi. Usa la parola chiave `as` per assegnare un alias a un campo di log o un risultato. Puoi utilizzare più alias in una query. Puoi utilizzare gli alias nei seguenti comandi:

- `fields`
- `parse`
- `sort`
- `stats`

Negli esempi seguenti viene illustrato come creare query che contengono alias.

Esempio

La query contiene un alias nel comando `fields`.

```
fields @timestamp, @message, accountId as ID
| sort @timestamp desc
| limit 20
```

La query restituisce i valori per i campi `@timestamp`, `@message` e `accountId`. I risultati sono ordinati in ordine decrescente e sono limitati a 20. I valori per `accountId` sono elencati sotto l'alias `ID`.

Esempio

La query contiene alias nei comandi `sort` e `stats`.

```
stats count(*) by duration as time
| sort time desc
```

La query conta il numero di volte che il campo `duration`è presente nel gruppo di log e ordina i risultati in ordine decrescente. I valori per `duration` sono elencati sotto l'alias `time`.

Utilizzo di commenti

CloudWatch Logs Insights supporta i commenti nelle query. Utilizza il carattere cancelletto (#) per impostare i commenti. Puoi utilizzare i commenti per ignorare le righe nelle query o nelle query di documenti.

Esempio: query

Quando viene emessa la seguente query, la seconda riga viene ignorata.

```
fields @timestamp, @message, accountId
# | filter accountId not like "7983124201998"
| sort @timestamp desc
| limit 20
```

Query di esempio

Questa sezione contiene un elenco di comandi di query generali e utili che è possibile eseguire nella console. CloudWatch Per informazioni su come eseguire un comando di query, consulta Tutorial: Esegui e modifica una query di esempio nella Amazon CloudWatch Logs User Guide.

Argomenti

- Query generali
- Query per i registri di Lambda

- Query per i flussi di log Amazon VPC
- Query per i registri di Route 53
- Interrogazioni per i log CloudTrail
- Interrogazioni per Amazon API Gateway
- Query per il gateway NAT
- Query per i registri del server Apache
- Interrogazioni per Amazon EventBridge
- Esempi del comando parse

Query generali

Trova i 25 log eventi aggiunti più di recente.

```
fields @timestamp, @message | sort @timestamp desc | limit 25
```

Ottieni un elenco del numero di eccezioni all'ora.

```
filter @message like /Exception/
    | stats count(*) as exceptionCount by bin(1h)
    | sort exceptionCount desc
```

Ottieni un elenco di log eventi che non sono eccezioni.

```
fields @message | filter @message not like /Exception/
```

Ottieni il log eventi più recente per ogni valore univoco del campo **`server`**.

```
fields @timestamp, server, severity, message
| sort @timestamp asc
| dedup server
```

Ottieni il log eventi più recente per ogni valore univoco del campo **`server`** per ogni tipo di **`severity`**.

```
fields @timestamp, server, severity, message
| sort @timestamp desc
```

```
| dedup server, severity
```

Query per i registri di Lambda

Determina la quantità di memoria per la quale è stato effettuato un provisioning eccessivo.

```
filter @type = "REPORT"
    | stats max(@memorySize / 1000 / 1000) as provisonedMemoryMB,
        min(@maxMemoryUsed / 1000 / 1000) as smallestMemoryRequestMB,
        avg(@maxMemoryUsed / 1000 / 1000) as avgMemoryUsedMB,
        max(@maxMemoryUsed / 1000 / 1000) as maxMemoryUsedMB,
        provisonedMemoryMB - maxMemoryUsedMB as overProvisionedMB
```

Crea un report sulla latenza.

```
filter @type = "REPORT" |
    stats avg(@duration), max(@duration), min(@duration) by bin(5m)
```

Cerca le invocazioni lente delle funzioni ed elimina le richieste duplicate che possono derivare da nuovi tentativi o dal codice lato client. In questa query, **`@duration`** è espresso in millisecondi.

```
fields @timestamp, @requestId, @message, @logStream
| filter @type = "REPORT" and @duration > 1000
| sort @timestamp desc
| dedup @requestId
| limit 20
```

Query per i flussi di log Amazon VPC

Trova i primi 15 trasferimenti di pacchetti tra gli host:

```
stats sum(packets) as packetsTransferred by srcAddr, dstAddr
    | sort packetsTransferred  desc
    | limit 15
```

Trova i primi 15 trasferimenti di byte per gli host su una determinata sottorete.

```
filter isIpv4InSubnet(srcAddr, "192.0.2.0/24")
```

```
    | stats sum(bytes) as bytesTransferred by dstAddr
    | sort bytesTransferred desc
    | limit 15
```

Trova gli indirizzi IP che utilizzano UDP come protocollo di trasferimento dei dati.

```
filter protocol=17 | stats count(*) by srcAddr
```

Trova gli indirizzi IP in cui i record di flusso sono stati ignorati durante la finestra di acquisizione.

```
filter logStatus="SKIPDATA"
    | stats count(*) by bin(1h) as t
    | sort t
```

Trova un singolo record per ogni connessione, per risolvere i problemi di connettività di rete.

```
fields @timestamp, srcAddr, dstAddr, srcPort, dstPort, protocol, bytes
| filter logStream = 'vpc-flow-logs' and interfaceId = 'eni-0123456789abcdef0'
| sort @timestamp desc
| dedup srcAddr, dstAddr, srcPort, dstPort, protocol
| limit 20
```

Query per i registri di Route 53

Trova la distribuzione di record all'ora in base al tipo di query.

```
stats count(*) by queryType, bin(1h)
```

Trova i 10 resolver DNS con il più elevato numero di richieste.

```
stats count(*) as numRequests by resolverIp
    | sort numRequests desc
    | limit 10
```

Trova il numero di record in base a dominio e sottodominio in cui il server non è riuscito a completare la richiesta DNS.

```
filter responseCode="SERVFAIL" | stats count(*) by queryName
```

Interrogazioni per i log CloudTrail

Trova il numero di voci di log per ogni servizio, tipo di evento e Regione AWS .

```
stats count(*) by eventSource, eventName, awsRegion
```

Trova gli host Amazon EC2 che sono stati avviati o interrotti in una determinata AWS regione.

```
filter (eventName="StartInstances" or eventName="StopInstances") and awsRegion="us-
east-2"
```

Trova le AWS regioni, i nomi utente e gli ARN degli utenti IAM appena creati.

```
filter eventName="CreateUser"
    | fields awsRegion, requestParameters.userName, responseElements.user.arn
```

Trova il numero di record in cui si è verificata un'eccezione durante il richiamo dell'API **`UpdateTrail`**.

```
filter eventName="UpdateTrail" and ispresent(errorCode)
    | stats count(*) by errorCode, errorMessage
```

Trova le voci di log in cui è stato utilizzato TLS 1.0 o 1.1

```
filter tlsDetails.tlsVersion in [ "TLSv1", "TLSv1.1" ]
| stats count(*) as numOutdatedTlsCalls by userIdentity.accountId, recipientAccountId,
 eventSource, eventName, awsRegion, tlsDetails.tlsVersion, tlsDetails.cipherSuite,
 userAgent
| sort eventSource, eventName, awsRegion, tlsDetails.tlsVersion
```

Trova il numero di chiamate per servizio che ha utilizzato le versioni TLS 1.0 o 1.1

```
filter tlsDetails.tlsVersion in [ "TLSv1", "TLSv1.1" ]
```

```
| stats count(*) as numOutdatedTlsCalls by eventSource
| sort numOutdatedTlsCalls desc
```

Interrogazioni per Amazon API Gateway

Trova gli ultimi 10 errori 4XX

```
fields @timestamp, status, ip, path, httpMethod
| filter status>=400 and status<=499
| sort @timestamp desc
| limit 10
```

Identifica le 10 Amazon API Gateway richieste che richiedono più tempo nel tuo gruppo di log di accesso Amazon API Gateway

```
fields @timestamp, status, ip, path, httpMethod, responseLatency
| sort responseLatency desc
| limit 10
```

Restituisci l'elenco dei percorsi API più popolari nel tuo Amazon API Gateway gruppo di log di accesso

```
stats count(*) as requestCount by path
| sort requestCount desc
| limit 10
```

Crea un rapporto sulla latenza di integrazione per il tuo gruppo di log di Amazon API Gateway accesso

```
filter status=200
| stats avg(integrationLatency), max(integrationLatency),
min(integrationLatency) by bin(1m)
```

Query per il gateway NAT

Se noti costi superiori al normale nella tua AWS fattura, puoi utilizzare CloudWatch Logs Insights per trovare i principali contributori. Per ulteriori informazioni sui seguenti comandi di query, vedi Come

posso trovare i principali contributori al traffico attraverso il gateway NAT nel mio VPC? nella pagina di supporto AWS premium.

> **Note**
>
> Nei seguenti comandi di query, sostituisci "x.x.x.x" con l'IP privato del gateway NAT e "y.y" con i primi due ottetti dell'intervallo CIDR VPC.

Trova le istanze che inviano più traffico attraverso il gateway NAT.

```
filter (dstAddr like 'x.x.x.x' and srcAddr like 'y.y.')
| stats sum(bytes) as bytesTransferred by srcAddr, dstAddr
| sort bytesTransferred desc
| limit 10
```

Determina il traffico da e verso le istanze nei gateway NAT.

```
filter (dstAddr like 'x.x.x.x' and srcAddr like 'y.y.') or (srcAddr like 'xxx.xx.xx.xx'
 and dstAddr like 'y.y.')
| stats sum(bytes) as bytesTransferred by srcAddr, dstAddr
| sort bytesTransferred desc
| limit 10
```

Determina le destinazioni Internet con cui le istanze del VPC comunicano più spesso per caricamenti e download.

Per i caricamenti

```
filter (srcAddr like 'x.x.x.x' and dstAddr not like 'y.y.')
| stats sum(bytes) as bytesTransferred by srcAddr, dstAddr
| sort bytesTransferred desc
| limit 10
```

Per i download

```
filter (dstAddr like 'x.x.x.x' and srcAddr not like 'y.y.')
| stats sum(bytes) as bytesTransferred by srcAddr, dstAddr
| sort bytesTransferred desc
| limit 10
```

Query per i registri del server Apache

È possibile utilizzare CloudWatch Logs Insights per interrogare i log del server Apache. Per ulteriori informazioni sulle seguenti domande, consulta Semplificazione dei log del server Apache con CloudWatch Logs Insights nel blog Cloud Operations & Migrations. AWS

Trova i campi più pertinenti in modo da poter rivedere i log di accesso e verificare la presenza di traffico nel percorso /admin dell'applicazione.

```
fields @timestamp, remoteIP, request, status, filename| sort @timestamp desc
| filter filename="/var/www/html/admin"
| limit 20
```

Trova il numero di richieste GET univoche che hanno effettuato l'accesso alla pagina principale con il codice di stato "200" (operazione riuscita).

```
fields @timestamp, remoteIP, method, status
| filter status="200" and referrer= http://34.250.27.141/ and method= "GET"
| stats count_distinct(remoteIP) as UniqueVisits
| limit 10
```

Trova il numero di volte in cui il servizio Apache è stato riavviato.

```
fields @timestamp, function, process, message
| filter message like "resuming normal operations"
| sort @timestamp desc
| limit 20
```

Interrogazioni per Amazon EventBridge

Ottieni il numero di EventBridge eventi raggruppati per tipo di dettaglio dell'evento

```
fields @timestamp, @message
| stats count(*) as numberOfEvents by `detail-type`
| sort numberOfEvents desc
```

Esempi del comando parse

Utilizza un'espressione glob per estrarre i campi **`@user`**, **`@method`** e **`@latency`** dal campo di log **`@message`** e restituire la latenza media per ogni combinazione univoca di **`@method`** e **`@user`**.

```
parse @message "user=*, method:*, latency := *" as @user,
    @method, @latency | stats avg(@latency) by @method,
    @user
```

Utilizza un'espressione regolare per estrarre i campi temporanei **@user2**, **@method2** e **@latency2** dal campo di log **@message** e restituire la latenza media per ogni combinazione univoca di **@method2** e **@user2**.

```
parse @message /user=(?<user2>.*?), method:(?<method2>.*?),
    latency := (?<latency2>.*?)/ | stats avg(latency2) by @method2,
    @user2
```

Estrae i campi **loggingTime**, **loggingType** e **loggingMessage**, filtra per log eventi che contengono le stringhe **ERROR** o **INFO** e quindi mostra solo i campi **loggingMessage** e **loggingType** per gli eventi che contengono una stringa **ERROR**.

```
FIELDS @message
    | PARSE @message "* [*] *" as loggingTime, loggingType, loggingMessage
    | FILTER loggingType IN ["ERROR", "INFO"]
    | DISPLAY loggingMessage, loggingType = "ERROR" as isError
```

Visualizzazione dei dati di log nei grafici

È possibile utilizzare visualizzazioni come grafici a barre, grafici a linee e grafici ad area in pila per identificare in modo più efficiente i modelli nei dati di registro. CloudWatch Logs Insights genera visualizzazioni per le query che utilizzano la `stats` funzione e una o più funzioni di aggregazione. Per ulteriori informazioni, consulta stats.

Salva e riesegui le query di Logs Insights CloudWatch

Dopo aver creato una query, è possibile salvarla in modo da poterla eseguire di nuovo in un secondo momento. Le query salvate vengono conservate in una struttura di cartelle per mantenerle organizzate. Puoi salvare fino a 1000 query per regione per account.

Per salvare una query, è necessario accedere a un ruolo che dispone dell'autorizzazione `logs:PutQueryDefinition`. Per visualizzare un elenco di query salvate, è necessario accedere a un ruolo che dispone dell'autorizzazione `logs:DescribeQueryDefinitions`.

Per salvare una query

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione scegli Logs (Registri), quindi Logs Insights.
3. Nell'editor di query, crea una query.
4. Selezionare Salva.

   Se non vedi il pulsante Salva, devi passare al nuovo design della console CloudWatch Logs. A tale scopo:

   a. Nel pannello di navigazione, selezionare Log groups (Gruppi di log).
   b. Scegliere Try the new design (Prova il nuovo design).
   c. Nel riquadro di spostamento scegliere Insights (Informazioni dettagliate) e tornare alla fase 3 di questa procedura.
5. Immettere un nome per la query.
6. (Facoltativo) Scegliere una cartella in cui si desidera salvare la query. Selezionare Create new (Crea nuova) per creare una cartella. Se si crea una nuova cartella, è possibile utilizzare i caratteri barra (/) nel nome della cartella per definire una struttura di cartelle. Ad esempio, la denominazione di una nuova cartella **`folder-level-1/folder-level-2`** crea una cartella di livello superiore denominata **`folder-level-1`**, con un'altra cartella chiamata **`folder-level-2`** all'interno di tale cartella. La query viene salvata in **`folder-level-2`**.
7. (Facoltativo) Modificare i gruppi di log o il testo della query.
8. Selezionare Salva.

> **Tip**
>
> Puoi creare una cartella per le query salvate con`PutQueryDefinition`. Per creare una cartella per le query salvate, utilizza una barra (/) per anteporre al nome della query desiderata il nome della cartella: <*`folder-name`*>/<*`query-name`*>. Per ulteriori informazioni su questa azione, consulta PutQueryDefinition.

Per eseguire una query salvata

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione scegli Logs (Registri), quindi Logs Insights.

3. A destra, scegliere Query.
4. Selezionare la query dall'elenco Saved queries (Query salvate). Viene visualizzata nell'editor di query.
5. Scegli Esegui.

Per salvare una nuova versione di una query salvata

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione scegli Logs (Registri), quindi Logs Insights.
3. A destra, scegliere Query.
4. Selezionare la query dall'elenco Saved queries (Query salvate). Viene visualizzata nell'editor di query.
5. Modificare la query. Se è necessario eseguire la query per controllare il proprio lavoro, scegliere Run query (Esegui query).
6. Quando si è pronti per salvare la nuova versione, scegliere Actions (Operazioni), Save as (Salva con nome).
7. Immettere un nome per la query.
8. (Facoltativo) Scegliere una cartella in cui si desidera salvare la query. Selezionare Create new (Crea nuova) per creare una cartella. Se si crea una nuova cartella, è possibile utilizzare i caratteri barra (/) nel nome della cartella per definire una struttura di cartelle. Ad esempio, la denominazione di una nuova cartella **`folder-level-1/folder-level-2`** crea una cartella di livello superiore denominata **`folder-level-1`**, con un'altra cartella chiamata **`folder-level-2`** all'interno di tale cartella. La query viene salvata in **`folder-level-2`**.
9. (Facoltativo) Modificare i gruppi di log o il testo della query.
10. Selezionare Salva.

Per eliminare una query, è necessario accedere a un ruolo che dispone dell'autorizzazione `logs:DeleteQueryDefinition`.

Per modificare o eliminare una query salvata

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione scegli Logs (Registri), quindi Logs Insights.
3. A destra, scegliere Query.

4. Selezionare la query dall'elenco Saved queries (Query salvate). Viene visualizzata nell'editor di query.
5. Scegliere Actions (Operazioni), Edit (Modifica) oppure Actions (Operazioni), Delete (Elimina).

Aggiunta di query a pannello di controllo o esportazione dei risultati della query

Dopo aver eseguito una query, puoi aggiungere la query a una CloudWatch dashboard o copiare i risultati negli Appunti.

Le query aggiunte ai pannelli di controllo vengono eseguite ogni volta che carichi il pannello di controllo e ogni volta che il pannello di controllo viene aggiornato. Queste query vengono conteggiate ai fini del limite di 30 query simultanee di Logs Insights CloudWatch .

Per aggiungere i risultati delle query a un pannello di controllo

1. Apri la console all'indirizzo https://console.aws.amazon.com/cloudwatch/ CloudWatch .
2. Nel pannello di navigazione scegli Logs (Registri), quindi Logs Insights.
3. Scegli uno o più gruppi di log ed esegui una query.
4. Scegli Add to dashboard (Aggiungi a pannello di controllo).
5. Seleziona il pannello di controllo, oppure scegli Create new (Crea nuovo) per creare un pannello di controllo per i risultati delle query.
6. Seleziona il tipo di widget da utilizzare per i risultati della query.
7. Inserisci un nome per il widget.
8. Scegli Add to dashboard (Aggiungi a pannello di controllo).

Per copiare i risultati della query negli appunti o scaricare i risultati della query

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione scegli Logs (Registri), quindi Logs Insights.
3. Scegli uno o più gruppi di log ed esegui una query.
4. Scegli Export results (Esporta risultati), quindi scegli l'opzione desiderata.

Visualizzazione di query in esecuzione o cronologia delle query

Puoi visualizzare le query attualmente in corso, nonché la cronologia delle query recenti.

Le query attualmente in esecuzione includono quelle aggiunte a un pannello di controllo. Hai un limite di 30 query simultanee di CloudWatch Logs Insights per account, incluse le query aggiunte ai dashboard.

Per visualizzare la cronologia delle query recenti

1. Apri la console all'indirizzo https://console.aws.amazon.com/cloudwatch/. CloudWatch
2. Nel pannello di navigazione scegli Logs (Registri), quindi Logs Insights.
3. Scegli Cronologia, se stai utilizzando il nuovo design per la console CloudWatch Logs. Se utilizzi il vecchio design, scegli Actions (Operazioni), View query history for this account (Visualizza cronologia query per questo account).

   Viene visualizzato un elenco delle query recenti. Puoi eseguire di nuovo qualsiasi query selezionandola e scegliendo Run (Esegui).

   In Stato, viene visualizzato il CloudWatch messaggio Registri in corso per tutte le interrogazioni attualmente in esecuzione.

Crittografa i risultati delle interrogazioni con AWS Key Management Service

Per impostazione predefinita, CloudWatch Logs crittografa i risultati archiviati delle query di CloudWatch Logs Insights utilizzando il metodo di crittografia predefinito Logs lato server. CloudWatch Puoi invece scegliere di utilizzare una chiave per crittografare questi risultati AWS KMS . Se associ una AWS KMS chiave ai risultati della crittografia, CloudWatch Logs utilizza quella chiave per crittografare i risultati archiviati di tutte le query nell'account.

Se successivamente si dissocia una chiave dai risultati della query, CloudWatch Logs torna al metodo di crittografia predefinito per le query successive. Tuttavia, le query eseguite mentre la chiave era associata sono ancora crittografate con quella chiave. CloudWatch I log possono comunque restituire tali risultati dopo la dissociazione della chiave KMS, poiché CloudWatch i registri possono continuare a fare riferimento alla chiave. Tuttavia, se la chiave viene successivamente disabilitata,

CloudWatch Logs non è in grado di leggere i risultati della query che sono stati crittografati con quella chiave.

> **Important**
>
> CloudWatch Logs supporta solo chiavi KMS simmetriche. Non utilizzare una chiave asimmetrica per crittografare i risultati della query. Per ulteriori informazioni, consulta la sezione relativa all'uso di chiavi simmetriche e asimmetriche.

Limiti

- Per eseguire la procedura seguente, devi avere le seguenti autorizzazioni: `kms:CreateKey`, `kms:GetKeyPolicy` e `kms:PutKeyPolicy`.
- Dopo aver associato o dissociato una chiave dai risultati della query, possono essere necessari fino a cinque minuti per rendere effettiva l'operazione.
- Se CloudWatch revochi l'accesso dei log a una chiave associata o elimini una chiave KMS associata, i dati crittografati in Logs non possono più essere recuperati. CloudWatch
- Non puoi utilizzare la CloudWatch console per associare una chiave, devi utilizzare l'API o Logs. AWS CLI CloudWatch

Fase 1: Creare un AWS KMS key

Per creare una chiave KMS, utilizza il seguente comando create-key:

```
aws kms create-key
```

L'output contiene l'ID chiave e l'Amazon Resource Name (ARN) della chiave. Di seguito è riportato un output di esempio:

```
{
    "KeyMetadata": {
        "Origin": "AWS_KMS",
        "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "Description": "",
        "KeyManager": "CUSTOMER",
        "Enabled": true,
        "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
```

```
        "KeyUsage": "ENCRYPT_DECRYPT",
        "KeyState": "Enabled",
        "CreationDate": 1478910250.94,
        "Arn": "arn:aws:kms:us-west-2:123456789012:key/6f815f63-e628-448c-8251-
e40cb0d29f59",
        "AWSAccountId": "123456789012",
        "EncryptionAlgorithms": [
            "SYMMETRIC_DEFAULT"
        ]
    }
}
```

Fase 2: Impostazione delle autorizzazioni sulla chiave KMS

Per impostazione predefinita, tutte le chiavi KMS sono private. Solo il proprietario della risorsa può utilizzarla per crittografare e decrittare i dati. Tuttavia, il proprietario della risorsa può concedere ad altri utenti e risorse le autorizzazioni per accedere alla chiave. Con questo passaggio, si CloudWatch concede al servizio Logs l'autorizzazione principale a utilizzare la chiave. L'entità del servizio deve trovarsi nella stessa AWS regione in cui è memorizzata la chiave.

Come procedura ottimale, si consiglia di limitare l'uso della chiave solo agli AWS account specificati.

Innanzitutto, salva la politica predefinita per la tua chiave KMS `policy.json` utilizzando il seguente get-key-policycomando:

```
aws kms get-key-policy --key-id key-id --policy-name default --output text > ./
policy.json
```

Aprire il file `policy.json` in un editor di testo e aggiungere la sezione in grassetto da una delle seguenti istruzioni. Separare l'istruzione esistente dalla nuova istruzione con una virgola. Queste istruzioni utilizzano `Condition` sezioni per migliorare la sicurezza della AWS KMS chiave. Per ulteriori informazioni, consulta AWS KMS chiavi e contesto di crittografia.

La `Condition` sezione di questo esempio limita l'uso della AWS KMS chiave ai risultati della query CloudWatch Logs Insights nell'account specificato.

```
{
 "Version": "2012-10-17",
    "Id": "key-default-1",
    "Statement": [
        {
```

```
            "Sid": "Enable IAM User Permissions",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::account_ID:root"
            },
            "Action": "kms:*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "logs.region.amazonaws.com"
            },
            "Action": [
                "kms:Encrypt*",
                "kms:Decrypt*",
                "kms:ReEncrypt*",
                "kms:GenerateDataKey*",
                "kms:Describe*"
            ],
            "Resource": "*",
            "Condition": {
                "ArnEquals": {
                    "aws:SourceArn": "arn:aws:logs:region:account_ID:query-result:*"
                },
                "StringEquals": {
                    "aws:SourceAccount": "Your_account_ID"
                }
            }
        }
    ]
}
```

Infine, aggiungi la politica aggiornata utilizzando il seguente put-key-policycomando:

```
aws kms put-key-policy --key-id key-id --policy-name default --policy file://
policy.json
```

Fase 3: associazione di una chiave KMS ai risultati della query

Associazione della chiave KMS ai risultati della query nell'account

Utilizza il comando disassociate-kms-key come segue:

```
aws logs associate-kms-key --resource-identifier "arn:aws:logs:region:account-id:query-result:*" --kms-key-id "key-arn"
```

Fase 4: Dissociazione di una chiave dai risultati della query nell'account

Per dissociare la chiave KMS associata ai risultati della query, usa il seguente disassociate-kms-keycomando:

```
aws logs disassociate-kms-key --resource-identifier "arn:aws:logs:region:account-id:query-result:*"
```

Utilizzo di gruppi di log e flussi di log

Un flusso di log è una sequenza di log eventi che condividono la stessa origine. Ogni fonte separata di log in CloudWatch Logs costituisce un flusso di log separato.

Un gruppo di log è un gruppo di flussi di log che condividono le stesse impostazioni di conservazione, monitoraggio e controllo degli accessi. Puoi definire i gruppi di log e specificare quali flussi inserire in ciascun gruppo. Non vi è alcun limite al numero di flussi di log che possono appartenere a un gruppo di log.

Utilizza le procedure di questa sezione per lavorare con i gruppi di log e i flussi di log.

Crea un gruppo di log in Logs CloudWatch

Quando installi l'agente CloudWatch Logs su un'istanza Amazon EC2 utilizzando i passaggi nelle sezioni precedenti della CloudWatch Amazon Logs User Guide, il gruppo di log viene creato come parte di tale processo. Puoi anche creare un gruppo di log direttamente nella console. CloudWatch

Creazione di un gruppo di log

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione, selezionare Log groups (Gruppi di log).
3. Selezionare Actions (Operazioni) e scegliere Create log group (Crea gruppo di log).
4. Immettere un nome per il gruppo di log, quindi selezionare Create log group (Crea gruppo di log).

> **Tip**
>
> È possibile preferire i gruppi di flussi di log, nonché i dashboard e gli allarmi, dal Preferiti e recenti nel riquadro di navigazione. Nella colonna Visitati di recente, passare il mouse sul gruppo di log che si desidera impostare come preferito e scegliere il simbolo della stella accanto a esso.

Invio di log a un gruppo di log

CloudWatch Logs riceve automaticamente gli eventi di registro da diversi AWS servizi. È inoltre possibile inviare altri eventi di registro a CloudWatch Logs utilizzando uno dei seguenti metodi:

- CloudWatch agente: l' CloudWatch agente unificato può inviare sia le metriche che i log ai registri. CloudWatch Per informazioni sull'installazione e l'utilizzo dell' CloudWatch agente, consulta Collecting metrics and logs from Amazon EC2 Instances and On-Premises Server with the CloudWatch Agent nella Amazon User Guide. CloudWatch
- AWS CLIput-log-events—Carica batch di eventi di registro su Logs. CloudWatch
- A livello di codice: l'PutLogEventsAPI consente di caricare in modo programmatico batch di eventi di registro nei registri. CloudWatch

Visualizza i dati di registro inviati ai registri CloudWatch

È possibile visualizzare e scorrere i dati di registro in stream-by-stream base a quelli inviati a CloudWatch Logs dall'agente CloudWatch Logs. Puoi specificare l'intervallo di tempo dei dati di log da visualizzare.

Visualizzazione dati di log

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione, selezionare Log groups (Gruppi di log).
3. In Log Groups (Gruppi di log), seleziona il gruppo per visualizzare i flussi.
4. Nell'elenco dei gruppi di log, scegliere il nome del gruppo di log che si desidera visualizzare.
5. Nell'elenco dei flussi di log, scegliere il nome del flusso di log che si desidera visualizzare.
6. Per modificare la modalità di visualizzazione dei dati di log, procedi in uno dei seguenti modi:
   - Per espandere un singolo log eventi, scegliere la freccia accanto all'evento di log.
   - Per espandere tutti gli eventi di log e visualizzarli come testo normale, sopra l'elenco di eventi di log, seleziona Text (Testo).
   - Per filtrare gli eventi di log, digitare il filtro di ricerca desiderato nel campo di ricerca. Per ulteriori informazioni, consulta Creazione di parametri da log eventi mediante filtri.
   - Per visualizzare i dati di log per un intervallo di data e ora specificato, scegliere la freccia accanto alla data e all'ora, accanto al filtro di ricerca. Per specificare un intervallo di data e ora, scegliere Absolute (Assoluto). Per scegliere un numero predefinito di minuti, ore, giorni o settimane, selezionare Relative (Relativo). Inoltre, è possibile passare da UTC a fuso orario locale.

Utilizzo di Live Tail per visualizzare i log quasi in tempo reale

CloudWatch Logs Live Tail ti aiuta a risolvere rapidamente gli incidenti visualizzando un elenco in streaming dei nuovi eventi di registro man mano che vengono inseriti. Puoi visualizzare, filtrare ed evidenziare i log importati quasi in tempo reale, in modo da poter rilevare e risolvere rapidamente i problemi. Puoi filtrare i log in base ai termini specificati e anche evidenziare quelli che contengono termini specifici per in modo da poter trovare rapidamente ciò che stai cercando.

Le sessioni Live Tail comportano costi al minuto in base al tempo di utilizzo della sessione. Per ulteriori informazioni sui prezzi, consulta la scheda Logs in Amazon CloudWatch Pricing.

Avvio di una sessione Live Tail

Utilizzi la CloudWatch console per avviare una sessione Live Tail. La procedura seguente spiega come avviare una sessione Live Tail utilizzando l'opzione Live Tail nel riquadro di navigazione a sinistra. Puoi anche avviare sessioni Live Tail dalla pagina Log Groups o dalla pagina CloudWatch Logs Insights.

> **Note**
>
> Se utilizzi policy di protezione dei dati per mascherare i dati sensibili in un gruppo di log che stai visualizzando con Live Tail, i dati sensibili appaiono sempre mascherati nella sessione Live Tail. Per ulteriori informazioni sul mascheramento dei dati sensibili nei gruppi di log, consulta Incremento della protezione dei dati di log sensibili con il mascheramento.

Per avviare una sessione Live Tail

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel riquadro di navigazione, scegli Log, Live Tail.
3. In Seleziona gruppi di log, seleziona i gruppi di log di cui desideri visualizzare gli eventi, nella sessione Live Tail. Puoi selezionare fino a 10 gruppi di log.
4. (Facoltativo) Se hai selezionato un solo gruppo di log, puoi filtrare ulteriormente la tua sessione Live Tail selezionando uno o più flussi di log da cui visualizzare i log eventi. A tale scopo, in Seleziona flussi di log, seleziona i nomi dei flussi di log dall'elenco a discesa. In alternativa, puoi utilizzare la seconda casella in Seleziona flussi di log per inserire il prefisso del nome del flusso di log, quindi vengono selezionati tutti i flussi di log con nomi che corrispondono al prefisso.

5. (Facoltativo) Per visualizzare solo i log eventi che contengono determinate parole o altre stringhe, inserisci la parola o la stringa in `Add filter patterns`.

   Ad esempio, per visualizzare solo i log eventi che includono la parola `Warning`, inserisci **`Warning`**. I filtri fanno distinzione tra maiuscole e minuscole. Puoi includere più termini e operatori di modelli in questo campo:

   - **`error 404`** mostra solo i log eventi che includono sia `error` che `404`
   - **`?Error ?error`** mostra i log eventi che includono `Error` o `error`
   - **`-INFO`** mostra tutti i log eventi che non includono `INFO`
   - **`{ $.eventType = "UpdateTrail" }`** mostra tutti i log eventi JSON in cui il valore del campo del tipo di evento è `UpdateTrail`

   Puoi anche usare l'espressione regolare (regex) per filtrare:

   - **`%ERROR%`** utilizza la regex per visualizzare tutti i log eventi costituiti dalla parola chiave ERROR
   - **`{ $.names = %Steve% }`** utilizza la regex per visualizzare i log eventi JSON in cui Steve si trova nella proprietà `"name"`
   - **`[ w1 = %abc%, w2 ]`** utilizza la regex per visualizzare i log eventi delimitati da spazi in cui la prima parola è abc

   Per ulteriori informazioni sulla sintassi dei modelli di filtro, consulta la sezione Sintassi di filtri e modelli.

6. (Facoltativo) Per evidenziare alcuni dei log eventi visualizzati, inserisci un termine da cercare ed evidenziare in Live Tail. Inserisci i termini da evidenziare uno alla volta. Se aggiungi più termini da evidenziare, viene assegnato un colore diverso per indicare ogni termine. Un indicatore di evidenziazione viene visualizzato a sinistra di qualsiasi log eventi che contiene il termine specificato e viene visualizzato anche sotto il termine stesso quando si espande il log eventi nella finestra principale per visualizzarlo nella sua interezza.

   Puoi utilizzare il filtro insieme all'evidenziazione per risolvere rapidamente i problemi. Ad esempio, potresti filtrare gli eventi per visualizzare solo quelli che contengono `Error` e quindi evidenziare anche quelli che contengono `404`.

7. Per avviare la sessione, scegli Applica filtri

I log eventi corrispondenti vengono visualizzati nella finestra. Inoltre, vengono visualizzate le seguenti informazioni:

- Il timer mostra per quanto tempo la sessione Live Tail è stata attiva.
- eventi/sec mostra quanti log eventi importati al secondo corrispondono ai filtri impostati.
- Per evitare che la sessione scorra troppo velocemente perché molti eventi corrispondono ai filtri, nei CloudWatch registri potrebbero essere visualizzati solo alcuni eventi corrispondenti. In tal caso, la percentuale di eventi corrispondenti visualizzati sullo schermo viene mostrata in % visualizzata.

8. Per sospendere il flusso di eventi e analizzare ciò che è attualmente visualizzato, fai clic in un punto qualsiasi della finestra degli eventi.
9. Durante la sessione, puoi utilizzare le seguenti opzioni per visualizzare maggiori dettagli su ogni log eventi.

   - Per visualizzare l'intero testo di un log eventi nella finestra principale, scegli la freccia accanto al relativo log eventi.
   - Per visualizzare l'intero testo di un log eventi in una finestra laterale, scegli la lente di ingrandimento + accanto al relativo log eventi. Il flusso di eventi viene sospeso e viene visualizzata la finestra laterale.

     La visualizzazione del testo di un log eventi nella finestra laterale può essere utile per confrontarne il testo con altri eventi nella finestra principale.
10. Per arrestare la sessione Live Tail, scegli Arresta.
11. Per riavviare la sessione, utilizza facoltativamente il pannello Filtro per modificare i criteri di filtro e scegli Applica filtri. Quindi selezionare Start (Avvia).

Ricerca di dati di log utilizzando i modelli dei filtri

Puoi cercare i tuoi dati di log utilizzando la Sintassi del modello di filtro per filtri di parametri, filtri di sottoscrizione, filtri di log eventi e Live Tail. È possibile cercare tutti i flussi di log all'interno di un gruppo di log oppure utilizzando il AWS CLI è possibile cercare anche flussi di log specifici. Quando ciascuna ricerca è in esecuzione, restituisce fino alla prima pagina di dati disponibili e un token per recuperare la pagina successiva di dati o per continuare la ricerca. Se non ottieni alcun risultato, puoi continuare la ricerca.

Puoi impostare l'intervallo di tempo per cui desideri eseguire query per limitare l'ambito della ricerca. Puoi iniziare con un intervallo di dimensioni maggiori per individuare i punti in cui le linee di log interessati cadono e ridurre quindi l'intervallo di tempo per diminuire la visualizzazione dei log nell'intervallo di tempo interessato.

Inoltre puoi muoverti direttamente dai tuoi parametri estratti dai log ai log corrispondenti.

Se hai effettuato l'accesso a un account configurato come account di monitoraggio in modalità osservabile CloudWatch tra più account, puoi cercare e filtrare gli eventi di registro dagli account di origine collegati a questo account di monitoraggio. Per maggiori informazioni, consulta la sezione Osservabilità su più account di CloudWatch .

Ricerca di voci di log utilizzando la console

Puoi cercare voci di log che soddisfino un criterio specificato utilizzando la console.

Ricerca di voci di log utilizzando la console

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione, selezionare Log groups (Gruppi di log).
3. In Log Groups (Gruppi di log), seleziona il nome del gruppo di log contenente il flusso di log da cercare.
4. In Log Streams (Flussi di log), seleziona il nome del flusso di log da cercare.
5. In Eventi di log, immettere la sintassi del filtro da utilizzare.

Cercare tutte le voci di log per un intervallo di tempo utilizzando la console

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione, selezionare Log groups (Gruppi di log).
3. In Log Groups (Gruppi di log), seleziona il nome del gruppo di log contenente il flusso di log da cercare.
4. Scegliere Cerca gruppo di log.
5. In Eventi di log, selezionare l'intervallo di data e ora e immettere la sintassi del filtro.

Cerca nelle voci del registro utilizzando il AWS CLI

È possibile cercare le voci di registro che soddisfano un criterio specificato utilizzando AWS CLI.

Per cercare le voci di registro utilizzando il AWS CLI

Al prompt dei comandi, esegui il comando seguente filter-log-events. Utilizza `--filter-pattern` per limitare i risultati per modello del filtro specificato e `--log-stream-names` per limitare i risultati a determinati flussi di log.

```
aws logs filter-log-events --log-group-name my-group [--log-stream-
names LIST_OF_STREAMS_TO_SEARCH] [--filter-pattern VALID_METRIC_FILTER_PATTERN]
```

Per cercare le voci di registro in un determinato intervallo di tempo utilizzando il AWS CLI

Al prompt dei comandi, esegui il filter-log-eventscomando seguente:

```
aws logs filter-log-events --log-group-name my-group [--log-stream-
names LIST_OF_STREAMS_TO_SEARCH] [--start-time 1482197400000] [--end-
time 1482217558365] [--filter-pattern VALID_METRIC_FILTER_PATTERN]
```

Cambiare da parametri a log

Puoi accedere a specifiche voci di log da altre parti della console.

Per accedere dai widget del pannello di controllo ai log

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione seleziona Dashboards (Pannelli di controllo).
3. Seleziona un pannello di controllo.
4. Nel widget, scegli l'icona View logs (Visualizza log) e quindi scegli View logs in this time range (Visualizza log in questo intervallo di tempo). Se è presente più di un filtro parametri, selezionane uno dall'elenco. Se sono presenti più filtri di parametri che possiamo mostrare nell'elenco, scegli Più filtri di parametri e seleziona o cerca un filtro di parametro.

Accesso da parametri a log

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel riquadro di navigazione, seleziona Parametri.
3. Nel campo di ricerca nella scheda All metrics (Tutti i parametri), digitare il nome del parametro e premi Invio.

4. Seleziona uno o più parametri dai risultati della tua ricerca.
5. Scegli Actions (Azioni), View logs (Visualizza log). Se è presente più di un filtro parametri, selezionane uno dall'elenco. Se sono presenti più filtri di parametri che possiamo mostrare nell'elenco, scegli Più filtri di parametri e seleziona o cerca un filtro di parametro.

Risoluzione dei problemi

Search takes too long to complete (La ricerca richiede troppo tempo per essere completata)

Se si dispone di una notevole quantità di dati di log, la ricerca potrebbe richiedere molto tempo per essere completata. Per velocizzare la ricerca, è possibile eseguire le operazioni descritte di seguito:

- Se utilizzi il AWS CLI, puoi limitare la ricerca solo ai flussi di log che ti interessano. Ad esempio, se il tuo gruppo di log ha 1000 flussi di log, ma desideri visualizzare solo tre flussi di log che ritieni pertinenti, puoi utilizzare il AWS CLI per limitare la ricerca solo ai tre flussi di log all'interno del gruppo di log.
- Utilizza un intervallo di tempo più breve e più granulare, che riduce la quantità di dati da ricercare e velocizza la query.

Modifica la conservazione dei dati di registro in Logs CloudWatch

Per impostazione predefinita, i dati di registro vengono archiviati nei CloudWatch registri a tempo indeterminato. Tuttavia, puoi configurare per quanto tempo archiviare i dati di log in un gruppo di log. I dati che superano l'impostazione di conservazione corrente verranno eliminati. Puoi modificare la conservazione dei log per ciascun gruppo di log in qualsiasi momento.

> **Note**
>
> CloudWatch Logs non elimina immediatamente gli eventi di registro quando raggiungono l'impostazione di conservazione. In genere sono necessarie fino a 72 ore prima che gli eventi di log vengano eliminati, ma in rare situazioni potrebbe essere necessario più tempo. Ciò significa che se modifichi un gruppo di log per avere un'impostazione di conservazione più lunga quando contiene eventi di log che hanno superato la data di scadenza, ma non sono stati effettivamente eliminati, l'eliminazione di tali eventi di log richiederà fino a 72 ore dopo il raggiungimento della nuova data di conservazione. Per assicurarsi che i dati di log vengano eliminati in modo definitivo, mantieni un gruppo di log con l'impostazione di conservazione inferiore fino a quando non sono trascorse 72 ore dopo la fine del periodo

di conservazione precedente oppure è stata confermata l'eliminazione degli eventi di log precedenti.
Quando i log eventi raggiungono l'impostazione di conservazione, vengono contrassegnati per l'eliminazione. Una volta contrassegnati per l'eliminazione, non vengono più considerati nei costi di archiviazione, anche se vengono effettivamente eliminati solo in un secondo momento. Questi log eventi contrassegnati per l'eliminazione non sono inclusi quando si utilizza un'API per recuperare il valore `storedBytes` per vedere quanti byte sta archiviando un gruppo di log.

Modifica dell'impostazione di conservazione di log

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione, selezionare Log groups (Gruppi di log).
3. Individua il gruppo di log da aggiornare.
4. Nella colonna Expire Events After (Scadenza eventi dopo) per tale gruppo di log, seleziona l'impostazione di conservazione corrente, ad esempio Never Expire (Nessuna scadenza).
5. In Edit Retention (Modifica conservazione), in Retention (Conservazione), scegliere un valore di conservazione di log, quindi Ok.

Contrassegna i gruppi di log in Amazon CloudWatch Logs

Puoi assegnare i tuoi metadati ai gruppi di log che crei in Amazon CloudWatch Logs sotto forma di tag. Un tag è una coppia chiave-valore che definisci per un gruppo di log. L'uso dei tag è un modo semplice ma efficace per gestire AWS le risorse e organizzare i dati, compresi i dati di fatturazione.

> **Note**
>
> È possibile utilizzare i tag per controllare l'accesso alle risorse di CloudWatch Logs, inclusi i gruppi di log e le destinazioni. L'accesso ai flussi di log è controllato a livello di gruppo di log per via della relazione gerarchica tra i gruppi di log e i flussi di log. Per ulteriori informazioni sull'utilizzo dei tag per controllare l'accesso alle risorse, consulta Controllo dell'accesso alle risorse di Amazon Web Services utilizzando i tag.

Indice

Nozioni di base sui tag

È possibile utilizzare AWS CloudFormation l'API AWS CLI, o CloudWatch Logs, per completare le seguenti attività:

- aggiunta di tag a un gruppo di log al momento della creazione;
- aggiunta di tag a un gruppo di log esistente;
- elencazione di tag di un gruppo di log;
- eliminazione di tag da un gruppo di log.

Puoi utilizzare i tag per categorizzare i gruppi di log. Ad esempio, puoi categorizzarli in base a scopo, proprietario o ambiente. Poiché definisci una chiave e un valore per ogni tag, puoi creare un set di categorie personalizzate per soddisfare esigenze specifiche. Ad esempio, puoi definire un set di tag che consente di monitorare i gruppi di log per proprietario e applicazione associata. Di seguito sono riportati vari esempi di tag:

- Progetto: nome del progetto
- Proprietario: nome
- Scopo: test di carico
- Applicazione: nome dell'applicazione
- Ambiente: produzione

Monitoraggio dei costi mediante l'assegnazione di tag

Puoi utilizzare i tag per classificare e tenere traccia AWS dei costi. Quando applichi tag alle tue AWS risorse, inclusi i gruppi di log, il report sull'allocazione AWS dei costi include l'utilizzo e i costi aggregati per tag. Puoi applicare i tag che rappresentano categorie di business (come centri di costo, nomi di applicazioni o proprietari) per organizzare i costi tra più servizi. Per ulteriori informazioni,

consulta Utilizzo dei tag per l'allocazione dei costi ai fini dei report di fatturazione personalizzati nella AWS Billing User Guide (Guida per l'utente di Amazon API Gateway).

Limitazioni applicate ai tag

Ai tag si applicano le limitazioni seguenti.

Limitazioni di base

- Il numero massimo di tag per ogni gruppo di log è 50.
- I valori e le chiavi dei tag rispettano la distinzione tra maiuscole e minuscole.
- Non puoi cambiare o modificare i tag di un gruppo di log eliminato.

Limitazioni applicate alle chiavi di tag

- Ogni chiave di tag deve essere univoca. Se aggiungi un tag con una chiave già in uso, il nuovo tag sovrascrive la coppia chiave-valore esistente.
- Non puoi iniziare una chiave di tag con `aws:` perché questo prefisso è riservato all'uso di. AWS AWS crea tag che iniziano con questo prefisso per tuo conto, ma non puoi modificarli o eliminarli.
- Le chiavi di tag devono avere una lunghezza compresa tra 1 e 128 caratteri Unicode.
- Le chiavi di tag devono contenere i seguenti caratteri: lettere Unicode, cifre, spazio e i seguenti caratteri speciali: `_ . / = + - @`.

Limitazioni applicate ai valori dei tag

- I valori dei tag devono avere una lunghezza compresa tra 0 e 255 caratteri Unicode.
- I valori dei tag possono essere vuoti. In caso contrario, devono contenere i seguenti caratteri: lettere Unicode, cifre, spazio e i seguenti caratteri speciali: `_ . / = + - @`.

Etichettare i gruppi di log utilizzando il AWS CLI

Puoi aggiungere, elencare e rimuovere tag tramite AWS CLI. Per alcuni esempi, consultare la seguente documentazione:

create-log-group

Crea un gruppo di log. Puoi opzionalmente aggiungere tag quando al momento della creazione del gruppo di log.

tag-resource

Assegna uno o più tag (coppie chiave-valore) alla risorsa Logs specificata. CloudWatch

list-tags-for-resource

Visualizza i tag associati a una risorsa Logs. CloudWatch

untag-resource

Rimuove uno o più tag dalla risorsa CloudWatch Logs specificata.

Etichettatura dei gruppi di log utilizzando l' CloudWatch API Logs

È possibile aggiungere, elencare e rimuovere tag utilizzando l'API CloudWatch Logs. Per alcuni esempi, consultare la seguente documentazione:

CreateLogGroup

Crea un gruppo di log. Puoi opzionalmente aggiungere tag quando al momento della creazione del gruppo di log.

TagResource

Assegna uno o più tag (coppie chiave-valore) alla risorsa Logs specificata. CloudWatch

ListTagsForResource

Visualizza i tag associati a una risorsa Logs. CloudWatch

UntagResource

Rimuove uno o più tag dalla risorsa CloudWatch Logs specificata.

Crittografa i dati di registro in CloudWatch Logs utilizzando AWS Key Management Service

I dati dei gruppi di log sono sempre crittografati in CloudWatch Logs. Per impostazione predefinita, CloudWatch Logs utilizza la crittografia lato server per i dati di registro inattivi. In alternativa, è

possibile utilizzare AWS Key Management Service per questa crittografia. In tal caso, la crittografia viene eseguita utilizzando una chiave. AWS KMS L'utilizzo della crittografia AWS KMS è abilitato a livello di gruppo di log, associando una chiave KMS a un gruppo di log, al momento della creazione del gruppo di log o dopo la sua esistenza.

> **Important**
>
> CloudWatch I registri ora supportano il contesto di crittografia, utilizzando `kms:EncryptionContext:aws:logs:arn` come chiave e l'ARN del gruppo di log come valore per quella chiave. Se disponi di gruppi di log già crittografati con una chiave KMS e desideri limitare l'utilizzo della chiave a un singolo account e gruppo di log, è necessario assegnare una nuova chiave KMS che include una condizione nella policy IAM. Per ulteriori informazioni, consulta AWS KMS chiavi e contesto di crittografia.

Dopo aver associato una chiave KMS a un gruppo di log, tutti i nuovi dati importati per il gruppo di log saranno crittografati tramite questa chiave. Questi dati vengono archiviati in formato crittografato per tutto il periodo di conservazione. CloudWatch I registri decrittografano questi dati ogni volta che vengono richiesti. CloudWatch I log devono disporre delle autorizzazioni per la chiave KMS ogni volta che vengono richiesti dati crittografati.

Se successivamente si dissocia una chiave KMS da un gruppo di log, CloudWatch Logs crittografa i dati appena acquisiti utilizzando il metodo di crittografia predefinito Logs. CloudWatch Tutti i dati precedentemente importati che sono stati crittografati con la chiave KMS rimangono crittografati con la chiave KMS. CloudWatch I log possono ancora restituire quei dati dopo che la chiave KMS è stata dissociata, perché CloudWatch i log possono continuare a fare riferimento alla chiave. Tuttavia, se la chiave viene successivamente disabilitata, CloudWatch Logs non è in grado di leggere i log che sono stati crittografati con quella chiave.

> **Important**
>
> CloudWatch Logs supporta solo chiavi KMS simmetriche. Non utilizzare una chiave asimmetrica per crittografare i dati nei gruppi di log. Per ulteriori informazioni, consulta la sezione relativa all'uso di chiavi simmetriche e asimmetriche.

Limiti

- Per eseguire la procedura seguente, devi avere le seguenti autorizzazioni: `kms:CreateKey`, `kms:GetKeyPolicy` e `kms:PutKeyPolicy`.
- Dopo aver associato o dissociato una chiave da un gruppo di log, possono essere necessari fino a cinque minuti per rendere effettiva l'operazione.
- Se CloudWatch revochi l'accesso dei log a una chiave associata o elimini una chiave KMS associata, i dati crittografati in Logs non possono più essere recuperati. CloudWatch
- Non è possibile associare una chiave KMS a un gruppo di log utilizzando la console. CloudWatch

Passaggio 1: creare una chiave AWS KMS

Per creare una chiave KMS, utilizza il seguente comando create-key:

```
aws kms create-key
```

L'output contiene l'ID chiave e l'Amazon Resource Name (ARN) della chiave. Di seguito è riportato un output di esempio:

```
{
    "KeyMetadata": {
        "Origin": "AWS_KMS",
        "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "Description": "",
        "KeyManager": "CUSTOMER",
        "Enabled": true,
        "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "KeyState": "Enabled",
        "CreationDate": 1478910250.94,
        "Arn": "arn:aws:kms:us-west-2:123456789012:key/6f815f63-e628-448c-8251-
e40cb0d29f59",
        "AWSAccountId": "123456789012",
        "EncryptionAlgorithms": [
            "SYMMETRIC_DEFAULT"
        ]
    }
}
```

Fase 2: Impostazione delle autorizzazioni sulla chiave KMS

Per impostazione predefinita, tutte AWS KMS le chiavi sono private. Solo il proprietario della risorsa può utilizzarla per crittografare e decrittare i dati. Tuttavia, il proprietario della risorsa può concedere ad altri utenti e risorse le autorizzazioni per accedere alla chiave KMS. Con questo passaggio, si CloudWatch concede al servizio Logs l'autorizzazione principale a utilizzare la chiave. L'entità del servizio deve trovarsi nella stessa AWS regione in cui è archiviata la chiave KMS.

Come procedura ottimale, consigliamo di limitare l'uso della chiave KMS solo agli AWS account o ai gruppi di log specificati.

Innanzitutto, salva la politica predefinita per la tua chiave KMS `policy.json` utilizzando il seguente comando: get-key-policy

```
aws kms get-key-policy --key-id key-id --policy-name default --output text > ./
policy.json
```

Aprire il file `policy.json` in un editor di testo e aggiungere la sezione in grassetto da una delle seguenti istruzioni. Separare l'istruzione esistente dalla nuova istruzione con una virgola. Queste istruzioni utilizzano `Condition` le sezioni per migliorare la sicurezza della AWS KMS chiave. Per ulteriori informazioni, consulta AWS KMS chiavi e contesto di crittografia.

La sezione `Condition` in questo esempio limita la chiave a un singolo ARN del gruppo di log.

```
{
 "Version": "2012-10-17",
    "Id": "key-default-1",
    "Statement": [
        {
            "Sid": "Enable IAM User Permissions",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::Your_account_ID:root"
            },
            "Action": "kms:*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "logs.region.amazonaws.com"
```

```
            },
            "Action": [
                "kms:Encrypt*",
                "kms:Decrypt*",
                "kms:ReEncrypt*",
                "kms:GenerateDataKey*",
                "kms:Describe*"
            ],
            "Resource": "*",
            "Condition": {
                "ArnEquals": {
                    "kms:EncryptionContext:aws:logs:arn": "arn:aws:logs:region:account-
id:log-group:log-group-name"
                }
            }
        }
    ]
}
```

La sezione `Condition` di questo esempio limita l'utilizzo di chiave AWS KMS all'account specificato, ma può essere utilizzato per qualsiasi gruppo di log.

```
{
    "Version": "2012-10-17",
    "Id": "key-default-1",
    "Statement": [
        {
            "Sid": "Enable IAM User Permissions",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::Your_account_ID:root"
            },
            "Action": "kms:*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "logs.region.amazonaws.com"
            },
            "Action": [
                "kms:Encrypt*",
                "kms:Decrypt*",
```

```
                "kms:ReEncrypt*",
                "kms:GenerateDataKey*",
                "kms:Describe*"
            ],
            "Resource": "*",
            "Condition": {
                "ArnLike": {
                    "kms:EncryptionContext:aws:logs:arn": "arn:aws:logs:region:account-
id:*"
                }
            }
        }
    ]
}
```

Infine, aggiungi la politica aggiornata utilizzando il seguente put-key-policycomando:

```
aws kms put-key-policy --key-id key-id --policy-name default --policy file://
policy.json
```

Fase 3: Associazione di una chiave KMS con un gruppo di log

Puoi associare una chiave KMS a un gruppo di log al momento della sua creazione o successivamente.

Per scoprire se a un gruppo di log è già associata una chiave KMS, usa il seguente describe-log-groupscomando:

```
aws logs describe-log-groups --log-group-name-prefix "log-group-name-prefix"
```

Se l'output include un campo `kmsKeyId`, il gruppo di log è associato alla chiave visualizzata per il valore di tale campo.

Associazione di una chiave KMS a un gruppo di log al momento della creazione

Utilizza il comando create-log-group come segue:

```
aws logs create-log-group --log-group-name my-log-group --kms-key-id "key-arn"
```

Associazione di una chiave KMS a un gruppo di log esistente

Utilizza il comando associate-kms-key come segue:

```
aws logs associate-kms-key --log-group-name my-log-group --kms-key-id "key-arn"
```

Fase 4: Dissociazione di una chiave da un gruppo di log

Per dissociare la chiave KMS associata a un gruppo di log, usa il seguente comando: disassociate-kms-key

```
aws logs disassociate-kms-key --log-group-name my-log-group
```

AWS KMS chiavi e contesto di crittografia

Per migliorare la sicurezza delle AWS Key Management Service chiavi e dei gruppi di log crittografati, CloudWatch Logs ora inserisce gli ARN dei gruppi di log come parte del contesto di crittografia utilizzato per crittografare i dati di registro. Il contesto di crittografia è un insieme di coppie chiave-valore che vengono utilizzate come dati autenticati aggiuntivi. Il contesto di crittografia consente di utilizzare le condizioni delle policy IAM per limitare l'accesso alla AWS KMS chiave per AWS account e gruppo di log. Per ulteriori informazioni, consulta Contesto di crittografia ed Elementi delle policy IAM JSON: condizione.

Si consiglia di utilizzare chiavi KMS diverse per ciascuno dei gruppi di log crittografati.

Se disponi di un gruppo di log crittografato in precedenza e desideri modificare il gruppo di log in modo da utilizzare una nuova chiave KMS dal cliente che funziona solo per tale gruppo di log, completa la seguente procedura.

Conversione di un gruppo di log crittografato in modo da utilizzare una chiave KMS con una policy che lo limita a tale gruppo di log

1. Immettere il seguente comando per trovare l'ARN del chiave corrente del gruppo di log:

   ```
   aws logs describe-log-groups
   ```

   L'output include la seguente riga. Prendere nota dell'ARN. È necessario utilizzarlo nel passaggio 7.

   ```
   ...
   "kmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/01234567-89ab-
   cdef-0123-456789abcdef"
   ...
   ```

2. Immettere il seguente comando per creare una nuova KMS:

```
aws kms create-key
```

3. Immettere il comando seguente per salvare la policy della nuova chiave in un file `policy.json`:

```
aws kms get-key-policy --key-id new-key-id --policy-name default --output text > ./
policy.json
```

4. Utilizzare un editor di testo per aprire `policy.json` e aggiungere un'espressione `Condition` alla policy:

```
{
    "Version": "2012-10-17",
    "Id": "key-default-1",
    "Statement": [
        {
            "Sid": "Enable IAM User Permissions",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::ACCOUNT-ID:root"
            },
            "Action": "kms:*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "logs.region.amazonaws.com"
            },
            "Action": [
                "kms:Encrypt*",
                "kms:Decrypt*",
                "kms:ReEncrypt*",
                "kms:GenerateDataKey*",
                "kms:Describe*"
            ],
            "Resource": "*",
            "Condition": {
                "ArnLike": {
                    "kms:EncryptionContext:aws:logs:arn":
 "arn:aws:logs:REGION:ACCOUNT-ID:log-
group:LOG-GROUP-NAME"
```

```
                }
            }
        }
    ]
}
```

5. Immettere il seguente comando per aggiungere la policy aggiornata alla nuova chiave KMS:

```
aws kms put-key-policy --key-id new-key-ARN --policy-name default --policy file://
policy.json
```

6. Immettere il comando seguente per associare la policy al gruppo di log:

```
aws logs associate-kms-key --log-group-name my-log-group --kms-key-id new-key-ARN
```

CloudWatch I log ora crittografano tutti i nuovi dati utilizzando la nuova chiave.

7. Quindi, revocare tutte le autorizzazioni tranne `Decrypt` dalla vecchia chiave. Innanzitutto, immettere il seguente comando per recuperare la vecchia policy:

```
aws kms get-key-policy --key-id old-key-ARN --policy-name default --output text
 > ./policy.json
```

8. Utilizzare un editor di testo per aprire `policy.json` e rimuovere tutti i valori dall'elenco `Action`, ad eccezione di `kms:Decrypt*`

```
{
    "Version": "2012-10-17",
    "Id": "key-default-1",
    "Statement": [
        {
            "Sid": "Enable IAM User Permissions",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::Your_account_ID:root"
            },
            "Action": "kms:*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "logs.region.amazonaws.com"
```

```
            },
            "Action": [
                "kms:Decrypt*"
            ],
            "Resource": "*"
        }
    ]
}
```

9. Immettere il seguente comando per aggiungere la policy aggiornata alla vecchia chiave:

```
aws kms put-key-policy --key-id old-key-ARN --policy-name default --policy file://
policy.json
```

Incremento della protezione dei dati di log sensibili con il mascheramento

Puoi contribuire a proteggere i dati sensibili che vengono acquisiti da CloudWatch Logs utilizzando le politiche di protezione dei dati dei gruppi di log. Queste policy consentono di verificare e mascherare i dati sensibili che appaiono nei log eventi importati dai gruppi di log dell'account.

Quando crei una politica di protezione dei dati, per impostazione predefinita, i dati sensibili che corrispondono agli identificatori di dati che hai selezionato vengono mascherati in tutti i punti di uscita, inclusi CloudWatch Logs Insights, filtri metrici e filtri di abbonamento. Solo gli utenti che dispongono dell'autorizzazione IAM `logs:Unmask` possono visualizzare i dati non mascherati.

Puoi creare una policy di protezione dei dati per tutti i gruppi di log del tuo account e puoi anche crearne una per i singoli gruppi di log. Quando crei una policy per l'intero account, questa si applica sia ai gruppi di log esistenti che a quelli creati successivamente.

Se crei una policy di protezione dei dati per l'intero account e una per un singolo gruppo di log, entrambe le policy si applicano a tale gruppo di log. Tutti gli identificatori di dati gestiti specificati in entrambe le policy vengono verificati e mascherati in tale gruppo di log.

Ogni gruppo di log può avere solo una policy di protezione dei dati a livello di gruppo di log, ma tale policy può specificare molti identificatori di dati gestiti da verificare e mascherare. Il limite per una policy di protezione dei dati è di 30.720 caratteri.

> **Important**
>
> I dati sensibili vengono rilevati e mascherati quando vengono importati nel gruppo di log. Quando si imposta una policy di protezione dei dati, i log eventi importati nel gruppo di log prima di quel momento non vengono mascherati.

CloudWatch La protezione dei dati di Logs consente di sfruttare i modelli di pattern matching e di machine learning per rilevare dati sensibili. I criteri e le tecniche utilizzati sono denominati identificatori dei dati gestiti. Le tecniche sono in grado di rilevare un ampio elenco di tipi di dati sensibili per molti Paesi e aree geografiche, inclusi dati finanziari, informazioni personali di identificazione e dati sanitari protetti. Per alcuni tipi di dati, il rilevamento dipende anche dalla ricerca di determinate parole chiave in prossimità dei dati sensibili.

Viene emessa una metrica relativa al CloudWatch momento in cui vengono rilevati dati sensibili che corrispondono agli identificatori di dati selezionati. Questa è la LogEventsWithFindingsmetrica e viene emessa nello spazio dei nomi AWS/Logs. Puoi utilizzare questa metrica per creare CloudWatch allarmi e visualizzarla in grafici e dashboard. Le metriche emesse dalla protezione dei dati sono metriche distribuite gratuite. Per ulteriori informazioni sulle metriche a cui invia Logs, consulta CloudWatch . CloudWatch Monitoraggio dell'utilizzo con i parametri di CloudWatch

Ogni identificatore di dati gestito è progettato per rilevare un tipo specifico di dati sensibili, come numeri di carte di credito, chiavi di accesso AWS segrete o numeri di passaporto per un determinato paese o area geografica. Quando crei una policy di protezione dei dati, puoi configurarla in modo che utilizzi questi identificatori per analizzare i log importati dal gruppo di log ed esegua operazioni specifiche quando tali dati vengono rilevati.

CloudWatch La protezione dei dati dei registri è in grado di rilevare le seguenti categorie di dati sensibili utilizzando identificatori di dati gestiti:

- Credenziali, come chiavi private o AWS chiavi di accesso segrete
- Informazioni finanziarie, ad esempio i numeri di carte di credito
- Informazioni personali di identificazione (PII), ad esempio patenti di guida o codici fiscali
- Dati sanitari protetti (PHI), ad esempio numeri di identificazione medica e assistenza sanitaria
- Identificatori di dispositivo, ad esempio indirizzi IP o indirizzi MAC

Per informazioni dettagliate sui tipi di dati che puoi proteggere, consulta la sezione Tipi di dati che è possibile proteggere.

Indice

- Informazioni finanziarie
  - ARN degli identificatori di dati per i tipi di dati finanziari
- Dati sanitari protetti (PHI)
  - ARN degli identificatori di dati per i tipi di dati sanitari protetti (PHI)
- Informazioni personali di identificazione (PII)
  - Parole chiave per i numeri identificativi delle patenti di guida
  - Parole chiave per i numeri di carta d'identità
  - Parole chiave per i numeri di passaporto
  - Parole chiave per i numeri di identificazione dei contribuenti e i numeri di riferimento
  - ARN degli identificatori dei dati per le informazioni personali di identificazione (PII)

Informazioni sulle policy di protezione dei dati

Argomenti

Cosa sono le policy di protezione dei dati?

CloudWatch Logs utilizza le politiche di protezione dei dati per selezionare i dati sensibili da scansionare e le azioni da intraprendere per proteggere tali dati. Per selezionare i dati sensibili di interesse, si utilizzano identificatori di dati. CloudWatch Registra la protezione dei dati, quindi rileva i dati sensibili utilizzando l'apprendimento automatico e il pattern matching. Per agire sugli identificatori di dati trovati, è possibile definire operazioni di audit (verifica) e de-identify (deidentifica). Queste operazioni consentono di registrare i dati sensibili trovati (o non trovati) e di mascherare i dati sensibili quando vengono visualizzati i log eventi.

Come è strutturata una policy di protezione dei dati?

Come illustrato nella figura riportata di seguito, un documento relativo alla policy di protezione dei dati include questi elementi:

- Informazioni opzionali sulla policy nella parte superiore del documento
- Una dichiarazione che definisce le azioni di audit e di deidentifica

È possibile definire una sola politica di protezione dei dati per gruppo di CloudWatch log Logs. La policy di protezione dei dati può includere una o più dichiarazioni di rifiuto o deidentificazione, ma solo una dichiarazione di verifica.

Proprietà JSON per la policy di protezione dei dati

Una policy di protezione dei dati richiede le seguenti informazioni di base ai fini dell'identificazione:

- Name (Nome): nome della policy.
- Description (Descrizione): (facoltativo) la descrizione della policy.
- Version (Versione): la versione del linguaggio della policy. La versione corrente è 2021-06-01.
- Statement (Dichiarazione): l'elenco di dichiarazioni che specificano le operazioni della policy di protezione dei dati.

```
{
  "Name": "CloudWatchLogs-PersonalInformation-Protection",
  "Description": "Protect basic types of sensitive data",
  "Version": "2021-06-01",
  "Statement": [
        ...
  ]
}
```

Proprietà JSON per una dichiarazione di policy

Una dichiarazione di policy definisce il contesto di rilevamento per l'operazione di protezione dei dati.

- Sid: (facoltativo) l'identificatore della dichiarazione.
- DataIdentifier— I dati sensibili per i quali CloudWatch Logs deve eseguire la scansione. Ad esempio, nome, indirizzo o numero di telefono.
- Funzionamento: le azioni successive, Audit o De-identify. CloudWatch Logs esegue queste azioni quando rileva dati sensibili.

```
{
  ...
  "Statement": [
    {
      "Sid": "audit-policy",
```

```
      "DataIdentifier": [
        "arn:aws:dataprotection::aws:data-identifier/Address"
      ],
      "Operation": {
        "Audit": {
          "FindingsDestination": {}
        }
      }
    },
```

Proprietà JSON per un'operazione della dichiarazione di policy

Una dichiarazione di policy definisce una delle seguenti operazioni di protezione dei dati.

- Verifica: emette metriche e report sui risultati senza interrompere la registrazione. Le stringhe che corrispondono incrementano la LogEventsWithFindingsmetrica che CloudWatch Logs pubblica nello spazio dei nomi AWS/Logs in. CloudWatch Puoi utilizzare queste metriche per creare allarmi.

  Per un esempio di un report sui risultati della verifica, consulta Report sui risultati della verifica.

  Per ulteriori informazioni sulle metriche a cui invia Logs, consulta. CloudWatch CloudWatch Monitoraggio dell'utilizzo con i parametri di CloudWatch
- Deidentifica: maschera i dati sensibili senza interrompere la registrazione.

Autorizzazioni IAM necessarie per la creazione o l'uso di una policy di protezione dei dati

Per poter utilizzare le policy di protezione dei dati per i gruppi di log, è necessario disporre di determinate autorizzazioni, come mostrato nelle tabelle seguenti. Le autorizzazioni sono diverse per le policy di protezione dei dati a livello di account e per quelle che si applicano a un singolo gruppo di log.

Autorizzazioni necessarie per le policy di protezione dei dati a livello di account

> **Note**
>
> Se si esegue una di queste operazioni all'interno di una funzione Lambda, il ruolo di esecuzione Lambda e il limite delle autorizzazioni devono includere anche le seguenti autorizzazioni.

Operazione	Autorizzazione IAM necessaria	Risorsa
Crea una policy di protezione dei dati senza destinazioni di verifica	`logs:PutAccountPolicy`	*
	`logs:PutDataProtectionPolicy`	*
Crea una politica di protezione dei dati con CloudWatch Logs come destinazione di controllo	`logs:PutAccountPolicy`	*
	`logs:PutDataProtectionPolicy`	*
	`logs:CreateLogDelivery`	*
	`logs:PutResourcePolicy`	*
	`logs:DescribeResourcePolicies`	*
	`logs:DescribeLogGroups`	*
Crea una policy di protezione dei dati con Kinesis Data Firehose come destinazione di verifica	`logs:PutAccountPolicy`	*
	`logs:PutDataProtectionPolicy`	*
	`logs:CreateLogDelivery`	*
	`firehose:TagDeliveryStream`	`arn:aws:logs:::deliverystre`

Operazione	Autorizzazione IAM necessaria	Risorsa
		am/ *YOUR_DELIVERY_STREAM*
Crea una policy di protezione dei dati con Amazon S3 come destinazione di verifica	`logs:PutAccountPolicy`	*
	`logs:PutDataProtectionPolicy`	*
	`logs:CreateLogDelivery`	*
	`s3:GetBucketPolicy`	`arn:aws:s3:::` *YOUR_BUCKET*
	`s3:PutBucketPolicy`	`arn:aws:s3:::` *YOUR_BUCKET*
Smaschera i log eventi mascherati in un gruppo di log specificato	`logs:Unmask`	`arn:aws:logs:::log-group:*`
Visualizza una policy di protezione dei dati esistente	`logs:GetDataProtectionPolicy`	*
Elimina una policy di protezione dei dati	`logs:DeleteAccountPolicy`	*
	`logs:DeleteDataProtectionPolicy`	*

Se i log di controllo della protezione dei dati sono già stati inviati a una destinazione, le altre policy che inviano i log alla stessa destinazione richiedono solo le autorizzazioni `logs:PutDataProtectionPolicy` e `logs:CreateLogDelivery`.

Autorizzazioni necessarie per le policy di protezione dei dati per un singolo gruppo di log

> **Note**
>
> Se si esegue una di queste operazioni all'interno di una funzione Lambda, il ruolo di esecuzione Lambda e il limite delle autorizzazioni devono includere anche le seguenti autorizzazioni.

Operazione	Autorizzazione IAM necessaria	Risorsa
Crea una policy di protezione dei dati senza destinazioni di verifica	`logs:PutDataProtectionPolicy`	`arn:aws:logs:::log-group:` *YOUR_LOG_GROUP* `:*`

Crea una politica di protezione dei dati con CloudWatch Logs come destinazione di controllo	`logs:PutDataProtectionPolicy`
	`logs:CreateLogDelivery`
	`logs:PutResourcePolicy`
	`logs:DescribeResourcePolicies`
	`logs:DescribeLogGroups`
		`arn:aws:logs:::log-group:` *YOUR_LOG_GROUP* `:*`
		`*`
		`*`
		`*`
		`*`

Crea una policy di protezione dei dati con Kinesis Data Firehose come destinazione di verifica	`logs:PutDataProtectionPolicy` `logs:CreateLogDelivery`	`arn:aws:logs:::log-group:` *YOUR_LOG_GROUP* `:*` `*`

Operazione	Autorizzazione IAM necessaria	Risorsa
	`firehose:TagDeliveryStream`	`arn:aws:logs:::deliverystream/` *YOUR_DELIVERY_STREAM*

Creazione di una policy di protezione dei dati con Amazon S3 come destinazione di controllo	`logs:PutDataProtectionPolicy`
	
	`logs:CreateLogDelivery`
	
	`s3:GetBucketPolicy`
	
	`s3:PutBucketPolicy`
		`arn:aws:logs:::log-group:` *YOUR_LOG_GROUP* `:*`
		
		`*`
		
		`arn:aws:s3:::` *YOUR_BUCKET*
		
		`arn:aws:s3:::` *YOUR_BUCKET*

Smascheramento di eventi di log mascherati	`logs:Unmask`	`arn:aws:logs:::log-group:` *YOUR_LOG_GROUP* `:*`
Visualizzazione di una policy di protezione dei dati esistente	`logs:GetDataProtectionPolicy`	`arn:aws:logs:::log-group:` *YOUR_LOG_GROUP* `:*`
Elimina una policy di protezione dei dati	`logs:DeleteDataProtectionPolicy`	`arn:aws:logs:::log-group:` *YOUR_LOG_GROUP* `:*`

Se i log di controllo della protezione dei dati sono già stati inviati a una destinazione, le altre policy che inviano i log alla stessa destinazione richiedono solo le autorizzazioni `logs:PutDataProtectionPolicy` e `logs:CreateLogDelivery`.

Policy di protezione dei dati di esempio

La seguente policy di esempio consente a un utente di creare, visualizzare ed eliminare policy di protezione dei dati in grado di inviare i risultati del controllo a tutti e tre i tipi di destinazioni di controllo. Non consente all'utente di visualizzare dati non mascherati.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "YOUR_SID_1",
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogDelivery",
                "logs:PutResourcePolicy",
                "logs:DescribeLogGroups",
                "logs:DescribeResourcePolicies"
            ],
            "Resource": "*"
        },
        {
            "Sid": "YOUR_SID_2",
            "Effect": "Allow",
            "Action": [
                "logs:GetDataProtectionPolicy",
                "logs:DeleteDataProtectionPolicy",
                "logs:PutDataProtectionPolicy",
                "s3:PutBucketPolicy",
                "firehose:TagDeliveryStream",
                "s3:GetBucketPolicy"
            ],
            "Resource": [
                "arn:aws:firehose:::deliverystream/YOUR_DELIVERY_STREAM",
                "arn:aws:s3:::YOUR_BUCKET",
                "arn:aws:logs:::log-group:YOUR_LOG_GROUP:*"
            ]
        }
    ]
}
```

Creazione di una policy di protezione dei dati a livello di account

Puoi utilizzare la console o i AWS CLI comandi CloudWatch Logs per creare una politica di protezione dei dati per mascherare i dati sensibili per tutti i gruppi di log del tuo account. Questa operazione ha effetto sia sui gruppi di log correnti che su quelli creati successivamente.

> **Important**
>
> I dati sensibili vengono rilevati e mascherati quando vengono importati nel gruppo di log. Quando si imposta una policy di protezione dei dati, i log eventi importati nel gruppo di log prima di quel momento non vengono mascherati.

Argomenti

Console

Utilizzo della console per creare una policy di protezione dei dati a livello di account

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione scegli Impostazioni. Si trova quasi in fondo all'elenco.
3. Scegliere la scheda Log.
4. Scegli Configura.
5. Per Identificatori di dati, seleziona i tipi di dati che desideri verificare e mascherare per tutti i tuoi gruppi di log. Per individuare gli identificatori che ti interessano, digita il testo nella casella di selezione.

   Ti consigliamo di selezionare solo gli identificatori di dati pertinenti per i tuoi dati di log e la tua attività. La scelta di numerosi tipi di dati può portare a falsi positivi.

   Per informazioni dettagliate sui tipi di dati che puoi proteggere, consulta la sezione Tipi di dati che è possibile proteggere.
6. (Facoltativo) Scegli uno o più servizi a cui inviare i risultati della verifica. Anche se scegli di non inviare i risultati della verifica ad alcun servizio, i tipi di dati sensibili selezionati verranno comunque mascherati.

7. Scegli Activate data protection (Attiva la protezione dei dati).

AWS CLI

Da utilizzare AWS CLI per creare una politica di protezione dei dati

1. Utilizza un editor di testo per creare un file di policy denominato `DataProtectionPolicy.json`. Per informazioni sulla sintassi delle policy, consulta la sezione seguente.
2. Immetti il comando seguente:

```
aws logs put-account-policy \
--policy-name TEST_POLICY --policy-type "DATA_PROTECTION_POLICY" \
--policy-document file://policy.json \
--scope "ALL" \
--region us-west-2
```

Sintassi della politica di protezione dei dati per le AWS CLI nostre operazioni API

Quando crei una policy di protezione dei dati JSON da utilizzare in un AWS CLI comando o in un'operazione API, la policy deve includere due blocchi JSON:

- Il primo blocco deve includere sia una matrice `DataIdentifer` sia una proprietà `Operation` con un'operazione `Audit`. La matrice `DataIdentifer` elenca i tipi di dati sensibili che desideri mascherare. Per ulteriori informazioni sulle opzioni disponibili, consulta Tipi di dati che è possibile proteggere.

  La proprietà `Operation` con l'operazione `Audit` è necessaria per individuare i termini relativi ai dati sensibili. L'operazione `Audit` deve contenere un oggetto `FindingsDestination`. È possibile utilizzare tale oggetto `FindingsDestination` per elencare una o più destinazioni a cui inviare il report sui risultati della verifica. Se specifichi destinazioni come gruppi di log, stream Amazon Kinesis Data Firehose e bucket S3, tali risorse devono già esistere. Per un esempio di report sui risultati della verifica, consulta Report sui risultati della verifica.
- Il secondo blocco deve includere sia una matrice `DataIdentifer` sia una proprietà `Operation` con un'operazione `Deidentify`. La matrice `DataIdentifer` deve corrispondere esattamente alla matrice `DataIdentifer` nel primo blocco della policy.

La proprietà `Operation` con l'operazione `Deidentify` è ciò che maschera effettivamente i dati e deve contenere l'oggetto `"MaskConfig": {}`. L'oggetto `"MaskConfig": {}` deve essere vuoto.

Quello che segue è un esempio di policy di protezione dei dati che maschera gli indirizzi e-mail e le patenti di guida degli Stati Uniti.

```
{
    "Name": "data-protection-policy",
    "Description": "test description",
    "Version": "2021-06-01",
    "Statement": [{
            "Sid": "audit-policy",
            "DataIdentifier": [
                "arn:aws:dataprotection::aws:data-identifier/EmailAddress",
                "arn:aws:dataprotection::aws:data-identifier/DriversLicense-US"
            ],
            "Operation": {
                "Audit": {
                    "FindingsDestination": {
                        "CloudWatchLogs": {
                            "LogGroup": "EXISTING_LOG_GROUP_IN_YOUR_ACCOUNT,"
                        },
                        "Firehose": {
                            "DeliveryStream": "EXISTING_STREAM_IN_YOUR_ACCOUNT"
                        },
                        "S3": {
                            "Bucket": "EXISTING_BUCKET"
                        }
                    }
                }
            }
        },
        {
            "Sid": "redact-policy",
            "DataIdentifier": [
                "arn:aws:dataprotection::aws:data-identifier/EmailAddress",
                "arn:aws:dataprotection::aws:data-identifier/DriversLicense-US"
            ],
            "Operation": {
                "Deidentify": {
```

```
                    "MaskConfig": {}
                }
            }
        }
    ]
}
```

Creazione di una policy di protezione dei dati per un singolo gruppo di log

È possibile utilizzare la console o i AWS CLI comandi CloudWatch Logs per creare una policy di protezione dei dati per mascherare i dati sensibili.

È possibile assegnare una policy di protezione dei dati a ciascun gruppo di log. Ogni policy di protezione dei dati può verificare diversi tipi di informazioni. Ogni policy di protezione dei dati può includere un'istruzione di verifica.

Argomenti

- Console
- AWS CLI

Console

Utilizzo della console per creare una policy di protezione dei dati

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione a sinistra, scegli Log, Gruppi di log.
3. Scegli il nome del gruppo di log.
4. Scegli Actions (Operazioni), quindi scegli Create data protection policy (Crea policy di protezione dei dati).
5. Per Data identifiers (Identificatori di dati), seleziona i tipi di dati che desideri controllare e mascherare in questo gruppo di log. Per individuare gli identificatori che ti interessano, digita il testo nella casella di selezione.

   Ti consigliamo di selezionare solo gli identificatori di dati pertinenti per i tuoi dati di log e la tua attività. La scelta di numerosi tipi di dati può portare a falsi positivi.

   Per informazioni dettagliate sui tipi di dati che puoi proteggere, consulta la sezione Tipi di dati che è possibile proteggere.

6. (Facoltativo) Scegli uno o più servizi a cui inviare i risultati della verifica. Anche se scegli di non inviare i risultati della verifica ad alcun servizio, i tipi di dati sensibili selezionati verranno comunque mascherati.
7. Scegli Activate data protection (Attiva la protezione dei dati).

AWS CLI

Da utilizzare AWS CLI per creare una politica di protezione dei dati

1. Utilizza un editor di testo per creare un file di policy denominato `DataProtectionPolicy.json`. Per informazioni sulla sintassi delle policy, consulta la sezione seguente.
2. Immetti il comando seguente:

```
aws logs put-data-protection-policy --log-group-identifier "my-log-group" --policy-
document file:///Path/DataProtectionPolicy.json --region us-west-2
```

Sintassi della politica di protezione dei dati per le AWS CLI nostre operazioni API

Quando crei una policy di protezione dei dati JSON da utilizzare in un AWS CLI comando o in un'operazione API, la policy deve includere due blocchi JSON:

- Il primo blocco deve includere sia una matrice `DataIdentifer` sia una proprietà `Operation` con un'operazione `Audit`. La matrice `DataIdentifer` elenca i tipi di dati sensibili che desideri mascherare. Per ulteriori informazioni sulle opzioni disponibili, consulta Tipi di dati che è possibile proteggere.

  La proprietà `Operation` con l'operazione `Audit` è necessaria per individuare i termini relativi ai dati sensibili. L'operazione `Audit` deve contenere un oggetto `FindingsDestination`. È possibile utilizzare tale oggetto `FindingsDestination` per elencare una o più destinazioni a cui inviare il report sui risultati della verifica. Se specifichi destinazioni come gruppi di log, stream Amazon Kinesis Data Firehose e bucket S3, tali risorse devono già esistere. Per un esempio di report sui risultati della verifica, consulta Report sui risultati della verifica.
- Il secondo blocco deve includere sia una matrice `DataIdentifer` sia una proprietà `Operation` con un'operazione `Deidentify`. La matrice `DataIdentifer` deve corrispondere esattamente alla matrice `DataIdentifer` nel primo blocco della policy.

La proprietà `Operation` con l'operazione `Deidentify` è ciò che maschera effettivamente i dati e deve contenere l'oggetto `"MaskConfig": {}`. L'oggetto `"MaskConfig": {}` deve essere vuoto.

Quello che segue è un esempio di policy di protezione dei dati che maschera gli indirizzi e-mail e le patenti di guida degli Stati Uniti.

```
{
    "Name": "data-protection-policy",
    "Description": "test description",
    "Version": "2021-06-01",
    "Statement": [{
            "Sid": "audit-policy",
            "DataIdentifier": [
                "arn:aws:dataprotection::aws:data-identifier/EmailAddress",
                "arn:aws:dataprotection::aws:data-identifier/DriversLicense-US"
            ],
            "Operation": {
                "Audit": {
                    "FindingsDestination": {
                        "CloudWatchLogs": {
                            "LogGroup": "EXISTING_LOG_GROUP_IN_YOUR_ACCOUNT,"
                        },
                        "Firehose": {
                            "DeliveryStream": "EXISTING_STREAM_IN_YOUR_ACCOUNT"
                        },
                        "S3": {
                            "Bucket": "EXISTING_BUCKET"
                        }
                    }
                }
            }
        },
        {
            "Sid": "redact-policy",
            "DataIdentifier": [
                "arn:aws:dataprotection::aws:data-identifier/EmailAddress",
                "arn:aws:dataprotection::aws:data-identifier/DriversLicense-US"
            ],
            "Operation": {
                "Deidentify": {
```

```
                    "MaskConfig": {}
                }
            }
        }
    ]
}
```

Visualizzazione di dati senza mascheramento

Per visualizzare i dati senza mascheramento, un utente deve disporre dell'autorizzazione `logs:Unmask`. Gli utenti che dispongono di questa autorizzazione possono visualizzare i dati senza mascheramento nei seguenti modi:

- Quando visualizzi gli eventi in un flusso di log, scegli Display (Visualizza), Unmask (Rimuovi mascheramento).
- Utilizza una query CloudWatch Logs Insights che include il comando unmask (@message). La seguente query di esempio mostra i 20 log eventi più recenti nel flusso senza mascheramento:

  ```
  fields @timestamp, @message, unmask(@message)
  | sort @timestamp desc
  | limit 20
  ```

  Per ulteriori informazioni sui comandi CloudWatch Logs Insights, vedere. CloudWatch Sintassi delle query di Logs Insights
- Utilizzare un' FilterLogEventsoperazione GetLogEventsor con il `unmask` parametro.

La CloudWatchLogsFullAccesspolitica include l'`logs:Unmask`autorizzazione. Per concedere `logs:Unmask` a un utente che non lo ha CloudWatchLogsFullAccess, puoi allegare una policy IAM personalizzata a quell'utente. Per ulteriori informazioni, consulta la sezione Adding permissions to a user (console) (Aggiunta di autorizzazioni a un utente [console]).

Report sui risultati della verifica

Se configuri le politiche di controllo della protezione dei dati di CloudWatch Logs per scrivere report di controllo su CloudWatch Logs, Amazon S3 o Kinesis Data Firehose, questi report sui risultati sono simili all'esempio seguente. CloudWatch Logs scrive un rapporto sui risultati per ogni evento di registro che contiene dati sensibili.

```
{
    "auditTimestamp": "2023-01-23T21:11:20Z",
    "resourceArn": "arn:aws:logs:us-west-2:111122223333:log-group:/aws/lambda/
MyLogGroup:*",
    "dataIdentifiers": [
        {
            "name": "EmailAddress",
            "count": 2,
            "detections": [
                {
                    "start": 13,
                    "end": 26
                },
{
                    "start": 30,
                    "end": 43
                }
            ]
        }
    ]
}
```

I campi del report sono i seguenti:

- Il campo `resourceArn` visualizza il gruppo di log in cui sono stati trovati i dati sensibili.
- L'oggetto `dataIdentifiers` visualizza le informazioni sui risultati di un tipo di dati sensibili in corso di verifica.
- Il campo `name` identifica il tipo di dati sensibili riportati in questa sezione.
- Il campo `count` visualizza il numero di volte in cui questo tipo di dati sensibili viene visualizzato nel log eventi.
- I campi `start` e `end` mostrano dove appare ogni occorrenza dei dati sensibili nel log eventi in base al numero di caratteri.

L'esempio precedente mostra un report sulla ricerca di due indirizzi e-mail in un log eventi. Il primo indirizzo e-mail inizia al 13° carattere del log eventi e termina al 26° carattere. Il secondo indirizzo e-mail va dal 30° al 43° carattere. Anche se questo log eventi ha due indirizzi e-mail, il valore della metrica `LogEventsWithFindings` viene incrementato solo di uno, poiché tale metrica conta il numero di log eventi che contengono dati sensibili, non il numero di occorrenze dei dati sensibili.

Politica chiave richiesta per inviare i risultati dell'audit a un bucket protetto da AWS KMS

Puoi proteggere i dati in un bucket Amazon S3 abilitando la crittografia lato server con Amazon S3 Managed Keys (SSE-S3) o la crittografia lato server con chiavi KMS (SSE-KMS). Per ulteriori informazioni, consulta Protezione dei dati con la crittografia lato server nella Guida per l'utente di Amazon S3.

Se si sceglie di inviare gli esiti dell'audit a un bucket protetto con SSE-S3, non è richiesta alcuna configurazione aggiuntiva. Amazon S3 gestisce la chiave di crittografia.

Se gli esiti dell'audit vengono inviati a un bucket protetto con SSE-KMS è necessario aggiornare la policy della chiave per la chiave KMS in modo che l'account di distribuzione dei log possa scrivere nel bucket S3. Per ulteriori informazioni sulla politica chiave richiesta per l'uso con SSE-KMS, consulta Amazon S3 la Amazon CloudWatch Logs User Guide.

Tipi di dati che è possibile proteggere

Questa sezione contiene informazioni sui tipi di dati che è possibile proteggere in una politica di protezione dei dati di CloudWatch Logs e sui paesi e le aree geografiche pertinenti per ciascun tipo di dati.

Per alcuni tipi di dati sensibili, CloudWatch Logs data protection analizza le parole chiave in prossimità dei dati e trova una corrispondenza solo se trova quella parola chiave. Se una parola chiave contiene uno spazio, CloudWatch Logs data protection corrisponde automaticamente alle varianti di parola chiave che non contengono lo spazio o che contengono un carattere di sottolineatura (_) o un trattino (-) anziché lo spazio. In alcuni casi, CloudWatch Logs amplia o abbrevia anche una parola chiave per rispondere alle varianti più comuni della parola chiave.

Indice

- Dati sanitari protetti (PHI)
  - ARN degli identificatori di dati per i tipi di dati sanitari protetti (PHI)
- Informazioni personali di identificazione (PII)
  - Parole chiave per i numeri identificativi delle patenti di guida
  - Parole chiave per i numeri di carta d'identità
  - Parole chiave per i numeri di passaporto
  - Parole chiave per i numeri di identificazione dei contribuenti e i numeri di riferimento
  - ARN degli identificatori dei dati per le informazioni personali di identificazione (PII)

CloudWatch Registra gli identificatori di dati gestiti per i tipi di dati sensibili

Le tabelle seguenti elencano i tipi di credenziali, dispositivi, informazioni finanziarie, mediche e sanitarie protette (PHI) che CloudWatch Logs è in grado di rilevare utilizzando identificatori di dati gestiti. Questi dati si aggiungono a determinati tipi di dati che potrebbero anche essere considerati informazioni personali di identificazione (PII).

Identificatori supportati indipendenti dalla lingua e dall'area geografica

Identificatore	Categoria
`Address`	Personale
`AwsSecretKey`	Credenziali
`CreditCardExpiration`	Servizi finanziari
`CreditCardNumber`	Servizi finanziari
`CreditCardSecurityCode`	Servizi finanziari
`EmailAddress`	Personale
`IpAddress`	Personale
`LatLong`	Personale
`Name`	Personale

Identificatore	Categoria
`OpenSshPrivateKey`	Credenziali
`PgpPrivateKey`	Credenziali
`PkcsPrivateKey`	Credenziali
`PuttyPrivateKey`	Credenziali
`VehicleIdentificationNumber`	Personale

Gli identificatori di dati dipendenti dall'area geografica prevedono il nome dell'identificatore seguito da un trattino e dai codici a due lettere (ISO 3166-1 alpha-2). Ad esempio, `DriversLicense-US`.

Identificatori supportati che devono includere un codice di Paese o area geografica a due lettere

Identificatore	Categoria	Paesi e lingue
BankAccountNumber	Servizi finanziari	DE, ES, FR, GB, IT
CepCode	Personale	BR
Cnpj	Personale	BR
CpfCode	Personale	BR
DriversLicense	Personale	AT, AU, BE, BG, CA, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IT, LT, LU, LV, MT, NL, PL, PT, RO, SE, SI, SK, US
DrugEnforcementAgencyNumber	Integrità	US
ElectoralRollNumber	Personale	GB
HealthInsuranceCardNumber	Integrità	UE

Identificatore	Categoria	Paesi e lingue
HealthInsuranceClaimNumber	Integrità	US
HealthInsuranceNumber	Integrità	FR
HealthcareProcedureCode	Integrità	US
IndividualTaxIdentification Number	Personale	US
InseeCode	Personale	FR
MedicareBeneficiaryNumber	Integrità	US
NationalDrugCode	Integrità	US
NationalIdentificationNumber	Personale	DE, ES, IT
NationalInsuranceNumber	Personale	GB
NationalProviderId	Integrità	US
NhsNumber	Integrità	GB
NieNumber	Personale	ES
NifNumber	Personale	ES
PassportNumber	Personale	CA, DE, ES, FR, GB, IT, US
PermanentResidenceNumber	Personale	CA
PersonalHealthNumber	Integrità	CA
PhoneNumber	Personale	BR, DE, ES, FR, GB, IT, US
PostalCode	Personale	CA
RgNumber	Personale	BR
SocialInsuranceNumber	Personale	CA

Identificatore	Categoria	Paesi e lingue
Ssn	Personale	ES, US
TaxId	Personale	DE, ES, FR, GB
ZipCode	Personale	US

Credenziali

CloudWatch Logs Data Protection consente di trovare i seguenti tipi di credenziali.

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e Regioni
AWS chiave di accesso segreta	`AwsSecretKey`	`aws_secret_access_key` , `credentials` , `secret access key`, `secret key`, `set-awscredential`	Tutti
Chiave privata OpenSSH	`OpenSSHPrivateKey`	Nessuno	Tutti
Chiave privata PGP	`PgpPrivateKey`	Nessuno	Tutti
Chiave privata Pkcs	`PkcsPrivateKey`	Nessuno	Tutti
Chiave privata PuTTY	`PuttyPrivateKey`	Nessuno	Tutti

ARN degli identificatori di dati per i tipi di dati credenziali

Di seguito sono elencati i nomi delle risorse Amazon (ARN) per gli identificatori dei dati che è possibile utilizzare per le policy di protezione dei dati.

ARN degli identificatori dei dati delle credenziali
`arn:aws:dataprotection::aws:data-identifier/AwsSecretKey`
`arn:aws:dataprotection::aws:data-identifier/OpenSshPrivateKey`
`arn:aws:dataprotection::aws:data-identifier/PgpPrivateKey`
`arn:aws:dataprotection::aws:data-identifier/PkcsPrivateKey`
`arn:aws:dataprotection::aws:data-identifier/PuttyPrivateKey`

Identificatori di dispositivo

CloudWatch Logs Data Protection può trovare i seguenti tipi di identificatori di dispositivo.

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e regioni
Indirizzo IP	`IpAddress`	Nessuno	Tutti

ARN degli identificatori di dati per i tipi di dati dispositivo

Di seguito sono elencati i nomi delle risorse Amazon (ARN) per gli identificatori dei dati che è possibile utilizzare per le policy di protezione dei dati.

ARN di identificatore dei dati del dispositivo
`arn:aws:dataprotection::aws:data-identifier/IpAddress`

Informazioni finanziarie

CloudWatch Logs Data Protection consente di trovare i seguenti tipi di informazioni finanziarie.

Se imposti una politica di protezione dei dati, CloudWatch Logs cerca gli identificatori di dati specificati indipendentemente dalla geolocalizzazione in cui si trova il gruppo di log. Le informazioni nella colonna Paesi e aree geografiche di questa tabella indicano se è necessario aggiungere i codici

Paese a due lettere all'identificatore di dati per individuare le parole chiave appropriate per tali Paesi e aree geografiche.

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e Regioni	Note
Numero del conto bancario	`BankAccountNumber`	Sì. A Paesi diversi si applicano parole chiave diverse. Per maggiori dettagli, consulta la tabella Parole chiave per i numeri di conto bancario più avanti in questa sezione.	Francia, Germania, Italia, Regno Unito, Spagna	Sono inclusi i codici International Bank Account Numbers (IBAN) composti da un massimo di 34 caratteri alfanumerici, inclusi elementi come i codici Paese.
Data di scadenza della carta di credito	`CreditCardExpiration`	`exp d`, `exp m`, `exp y`, `expiration` , `expiry`	Tutti	
Numero di carta di credito	`CreditCardNumber`	`account number`, `american express`, `amex`, `bank card`, `card`, `card number`, `card`	Tutti	Il rilevamento richiede

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e Regioni	Note
		num, cc #, ccn, check card, credit, credit card#, dankort, debit, debit card, diners club, discover, electron, japanese card bureau, jcb, mastercard , mc, pan, payment account number, payment card number, pcn, union pay, visa		che i dati siano una sequenza di 13-19 cifre che rispetti la formula Luhn check e utilizzi un prefisso numerico di carta standard per uno dei seguenti tipi di carte di credito: American Express, Dankort,

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e Regioni	Note
				Diner's Club, Discover, Electron, Japanese Card Bureau (JCB), Mastercard e Visa. UnionPay
Codice di verifica della carta di credito	`CreditCardSecurityCode`	`card id`, `card identification code`, `card identification number` , `card security code`, `card validation code` , `card validation number` , `card verification data` , `card verification value`, `cvc`, `cvc2`, `cvv`, `cvv2`, `elo verification code`	Tutti	

Parole chiave per i numeri di conto bancario

Utilizza le seguenti parole chiavi per rilevare i codici International Bank Account Numbers (IBAN), composti da un massimo di 34 caratteri alfanumerici, inclusi elementi come i codici Paese.

Paese	Parole chiave
Francia	account code, account number, accountno# , accountnumber# , bban, code bancaire, compte bancaire, customer account id, customer account number, customer bank account id, iban, numéro de compte
Germania	account code, account number, accountno# , accountnumber# , bankleitzahl , bban, customer account id, customer account number, customer bank account id, geheimzahl , iban, kartennummer , kontonummer , kreditkartennummer , sepa
Italia	account code, account number, accountno# , accountnumber# , bban, codice bancario, conto bancario, customer account id, customer account number, customer bank account id, iban, numero di conto
Spagna	account code, account number, accountno# , accountnumber# , bban, código cuenta, código cuenta bancaria, cuenta cliente id, customer account ID, customer account number, customer bank account id, iban, número cuenta bancaria cliente, número cuenta cliente
Regno Unito	account code, account number, accountno# , accountnumber# , bban, customer account ID, customer account number, customer bank account id, iban, sepa

CloudWatch I registri non riportano le occorrenze delle seguenti sequenze, che gli emittenti di carte di credito hanno riservato ai test pubblici.

```
122000000000003, 2222405343248877, 2222990905257051, 2223007648726984,
 2223577120017656,
30569309025904, 34343434343434, 3528000700000000, 3530111333300000, 3566002020360505,
 36148900647913,
36700102000000, 371449635398431, 378282246310005, 378734493671000, 38520000023237,
 4012888888881881,
4111111111111111, 4222222222222, 4444333322221111, 4462030000000000, 4484070000000000,
 4911830000000,
```

```
4917300800000000, 4917610000000000, 4917610000000000003, 5019717010103742,
 5105105105105100,
5111010030175156, 5185540810000019, 5200828282828210, 5204230080000017,
 5204740009900014, 5420923878724339,
5454545454545454, 5455330760000018, 5506900490000436, 5506900490000444,
 5506900510000234, 5506920809243667,
5506922400634930, 5506927427317625, 5553042241984105, 5555553753048194,
 5555555555554444, 5610591081018250,
6011000990139424, 6011000400000000, 6011111111111117, 630490017740292441,
 630495060000000000,
6331101999990016, 6759649826438453, 6799990100000000019, and 76009244561.
```

ARN degli identificatori di dati per i tipi di dati finanziari

Di seguito sono elencati i nomi delle risorse Amazon (ARN) per gli identificatori dei dati che è possibile utilizzare per le policy di protezione dei dati.

ARN degli identificatore dei dati finanziari
arn:aws:dataprotection::aws:data-identifier/BankAccountNumber-DE
arn:aws:dataprotection::aws:data-identifier/BankAccountNumber-ES
arn:aws:dataprotection::aws:data-identifier/BankAccountNumber-FR
arn:aws:dataprotection::aws:data-identifier/BankAccountNumber-GB
arn:aws:dataprotection::aws:data-identifier/BankAccountNumber-IT
arn:aws:dataprotection::aws:data-identifier/BankAccountNumber-US
arn:aws:dataprotection::aws:data-identifier/CreditCardExpiration
arn:aws:dataprotection::aws:data-identifier/CreditCardNumber
arn:aws:dataprotection::aws:data-identifier/CreditCardSecurityC ode

Dati sanitari protetti (PHI)

CloudWatch Logs Data Protection può trovare i seguenti tipi di informazioni sanitarie protette (PHI).

Se imposti una politica di protezione dei dati, CloudWatch Logs cerca gli identificatori di dati specificati indipendentemente dalla geolocalizzazione in cui si trova il gruppo di log. Le informazioni nella colonna Paesi e aree geografiche di questa tabella indicano se è necessario aggiungere i codici Paese a due lettere all'identificatore di dati per individuare le parole chiave appropriate per tali Paesi e aree geografiche.

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e regioni
Numero di registrazione Drug Enforcement Agency (DEA)	`DrugEnforcementAgencyNumber`	`dea number`, `dea registration`	Stati Uniti
Numero EHIC (Health Insurance Card)	`HealthInsuranceCardNumber`	`assicurazione sanitaria numero`, `carta assicurazione numero`, `carte d’assurance maladie `, `carte européenne d'assurance maladie `, `ceam`, `ehic`, `ehic#`, `finlandehicnumber#  `, `gesundheitskarte `, `hälsokort `, `health card`, `health card number`, `health insurance card`, `health insurance number`, `insurance card number`, `krankenversicherungskarte `, `krankenversicherungsnummer  `, `medical account number`, `numero conto medico`, `numéro d’assurance maladie `, `numéro de carte d’assurance `, `numéro de compte`	Unione Europea

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e regioni
		medical, número de cuenta médica, número de seguro de salud, número de tarjeta de seguro, sairaanhoitokortin , sairausvakuutuskortti , sairausvakuutusnumero , sjukförsäkringnummer, sjukförsäkringskort , suomiehic-numero , tarjeta de salud, terveyskortti , tessera sanitaria assicurazione numero , versicherungsnummer	
Health Insurance Claim Number (HICN)	HealthInsuranceClaimNumber	health insurance claim number, hic no, hic no., hic number, hic#, hicn, hicn#, hicno#	Stati Uniti
Numero di identificazione medica e assistenza sanitaria	HealthInsuranceNumber	carte d'assuré social, carte vitale, insurance card	Francia
Codice HCPCS (Healthcare Common Procedure Coding System)	HealthcareProcedureCode	current procedural terminology , hcpcs, healthcare common procedure coding system	Stati Uniti

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e regioni
Numero MBN (Medicare Beneficiary Number)	`MedicareBeneficiaryNumber`	`mbi`, `medicare beneficiary`	Stati Uniti
National Drug Code (NDC)	`NationalDrugCode`	`national drug code`, `ndc`	Stati Uniti
National Provider Identifier (NPI)	`NationalProviderId`	`hipaa`, `n.p.i.`, `national provider`, `npi`	Stati Uniti
Numero National Health Service (NHS)	`NhsNumber`	`national health service`, `NHS`	Gran Bretagna
Personal Health Number	`PersonalHealthNumber`	`canada healthcare number`, `msp number`, `care number`, `phn`, `soins de santé`	Canada

ARN degli identificatori di dati per i tipi di dati sanitari protetti (PHI)

Di seguito sono elencati i nomi delle risorse Amazon (ARN) per gli identificatori di dati possono essere utilizzati nelle policy di protezione dei dati sanitari protetti (PHI).

ARN per identificatori di dati sanitari protetti
`arn:aws:dataprotection::aws:data-identifier/DrugEnforcementAgencyNumber-US`
`arn:aws:dataprotection::aws:data-identifier/HealthcareProcedureCode-US`
`arn:aws:dataprotection::aws:data-identifier/HealthInsuranceCardNumber-EU`

ARN per identificatori di dati sanitari protetti
arn:aws:dataprotection::aws:data-identifier/HealthInsuranceClaimNumber-US
arn:aws:dataprotection::aws:data-identifier/HealthInsuranceNumber-FR
arn:aws:dataprotection::aws:data-identifier/MedicareBeneficiaryNumber-US
arn:aws:dataprotection::aws:data-identifier/NationalDrugCode-US
arn:aws:dataprotection::aws:data-identifier/NationalInsuranceNumber-GB
arn:aws:dataprotection::aws:data-identifier/NationalProviderId-US
arn:aws:dataprotection::aws:data-identifier/NhsNumber-GB
arn:aws:dataprotection::aws:data-identifier/PersonalHealthNumber-CA

Informazioni personali di identificazione (PII)

CloudWatch La protezione dei dati di Logs può trovare i seguenti tipi di informazioni di identificazione personale (PII).

Se imposti una politica di protezione dei dati, CloudWatch Logs cerca gli identificatori di dati specificati indipendentemente dalla geolocalizzazione in cui si trova il gruppo di log. Le informazioni nella colonna Paesi e aree geografiche di questa tabella indicano se è necessario aggiungere i codici Paese a due lettere all'identificatore di dati per individuare le parole chiave appropriate per tali Paesi e aree geografiche.

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e Regioni	Note
Data di nascita	`DateOfBirth`	`dob`, `date of birth`, `birthdate` , `birth date`, `birthday`, `b-day`, `bday`	Qualsias	Il supporto include la maggior parte dei formati di data, ad esempio tutte le cifre e le combinazioni di cifre e nomi dei mesi. I componenti della data possono essere separati da spazi, barre (/) o

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e Regioni	Note
				trattini (-).
Código de Endereçamento Postal (CEP)	`CepCode`	`cep`, `código de endereçamento postal`, `codigo de endereçamento postal`	Brasile	
Cadastro Nacional da Pessoa Jurídica (CNPJ)	`Cnpj`	`cadastro nacional da pessoa jurídica`, `cadastro nacional da pessoa juridica`, `cnpj`	Brasile	
Cadastro de Pessoas Físicas (CPF)	`CpfCode`	`Cadastro de pessoas fisicas`, `cadastro de pessoas físicas`, `cadastro de pessoa física`, `cadastro de pessoa fisica`, `cpf`	Brasile	

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e Regioni	Note
Numero identificativo della patente di guida	`DriversLicense`	Sì. A Paesi diversi si applicano parole chiave diverse. Per i dettagli, consulta la tabella Numeri identificativi della patente di guida più avanti in questa sezione.	Molti Paesi. Per i dettagli, consulta la tabella Numeri identificativi della patente di guida.	
Numero di lista elettorale	`ElectoralRollNumber`	`electoral #`, `electoral number`, `electoral roll #`, `electoral roll no.`, `electoral roll number`, `electoral rollno`	Regno Unito	
Identificazione del singolo contribuente	`IndividualTaxIdentificationNumber`	Sì. A Paesi diversi si applicano parole chiave diverse. Per i dettagli, consulta la tabella Numeri di identificazione del singolo contribuente più avanti in questa sezione.	Brasile, Francia, Germani Regno Unito, Spagna	

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e Regioni	Note
Institut national de la statistique et des études économiques (INSEE)	`InseeCode`	Sì. A Paesi diversi si applicano parole chiave diverse. Per i dettagli, consulta la tabella Parole chiave per i numeri di identificazione nazionale più avanti in questa sezione.	Francia	

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e Regioni	Note
Numero di identificazione nazionale	`NationalIdentificationNumber`	Sì. Per i dettagli, consulta la tabella Parole chiave per i numeri di identificazione nazionale più avanti in questa sezione.	Germani Italia, Spagna	Sono inclusi gli identificatori DNI (Documento Nacional de Identidad, Spagna), il codice fiscale (Italia) e i numeri di carta d'identità nazionale (Germania).

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e Regioni	Note
Numero NINO (National Insurance Number)	`NationalInsuranceNumber`	`insurance no.`, `insurance number`, `insurance#` , `national insurance number`, `nationalinsurance#` , `nationalinsurancenumber` , `nin`, `nino`	–	Regno Unito
Número de identidad de extranjero (NIE)	`NieNumber`	Sì. A Paesi diversi si applicano parole chiave diverse. Per i dettagli, consulta la tabella Numeri di identificazione del singolo contribuente più avanti in questa sezione.	Spagna	
Número de Identificación Fiscal (NIF)	`NifNumber`	Sì. A Paesi diversi si applicano parole chiave diverse. Per i dettagli, consulta la tabella Numeri di identificazione del singolo contribuente più avanti in questa sezione.	Spagna	

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e Regioni	Note
Numero di passaporto	`PassportNumber`	Sì. A Paesi diversi si applicano parole chiave diverse. Per maggiori dettagli, consulta la tabella Parole chiave per i numeri di passaporto più avanti in questa sezione.	Canada, Francia, Germani Italia, Regno Unito, Spagna, Stati Uniti	
Numero di residenza permanente (Green Card)	`PermanentResidenceNumber`	`carte résident permanent` , `numéro carte résident permanent` , `numéro résident permanent` , `permanent resident card`, `permanent resident card number`, `permanent resident no`, `permanent resident no.`, `permanent resident number`, `pr no`, `pr no.`, `pr non`, `pr number`, `résident permanent no.`, `résident permanent non`	Canada	

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e Regioni	Note
Numero di telefono	PhoneNumber	Brasile: le parole chiave comprendono anche: `cel`, `celular`, `fone`, `móvel`, `número residencial` , `numero residencial` , `telefone` Altri Paesi: `cell`, `contact`, `fax`, `fax number`, `mobile`, `phone`, `phone number`, `tel`, `telephone` , `telephone number`	Brasile, Canada, Francia, Germani Italia, Regno Unito, Spagna, Stati Uniti	Sono inclusi i numeri di fax e i numeri verdi negli Stati Uniti. Se una parola chiave si trova in prossimità dei dati, il numero non deve includere il prefisso internazionale. Se una parola chiave non

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e Regioni	Note
				è in prossimità dei dati, il numero deve includere un prefisso internazionale.
Codice postale	`PostalCode`	Nessuno	Canada	
Registro Geral (RG)	`RgNumber`	Sì. A Paesi diversi si applicano parole chiave diverse. Per i dettagli, consulta la tabella Numeri di identificazione del singolo contribuente più avanti in questa sezione.	Brasile	
Social Insurance Number (SIN)	`SocialInsuranceNumber`	`canadian id`, `numéro d'assurance sociale`, `social insurance number`, `sin`	Canada	

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e Regioni	Note
Social Security Number (SSN)	`Ssn`	Spagna: `número de la seguridad social`, `social security no.`, `social security no.` `número de la seguridad social`, `social security number`, `socialsecurityno#` , `ssn`, `ssn#` Stati Uniti: `social security`, `ss#`, `ssn`	Spagna, Stati Uniti	
Numero identificativo del contribuente o codice fiscale	`TaxId`	Sì. A Paesi diversi si applicano parole chiave diverse. Per i dettagli, consulta la tabella Numeri di identificazione del singolo contribuente più avanti in questa sezione. .	Francia, Germania, Regno Unito, Spagna	Sono inclusi: TIN (Francia); Steueridentifikationsnummer (Germania); CIF (Spagna) e TRN, UTR (Regno Unito).

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e Regioni	Note
Codice postale	`ZipCode`	`zip code`, `zip+4`	Stati Uniti	Codice postale degli Stati Uniti.
Indirizzo postale	`Address`	Nessuno	Australia, Canada, Francia, Germania, Italia, Regno Unito, Spagna, Stati Uniti	Sebbene non sia richiesta una parola chiave, il rilevamento richiede che l'indirizzo includa il nome di una città o di un luogo e un CAP o codice postale.

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e Regioni	Note
Indirizzo e-mail	`EmailAddress`	Nessuno	Qualsias	

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e Regioni	Note
Coordinate del sistema di posizionamento globale (GPS)	`LatLong`	`coordinate` , `coordinates` , `latlong`, `latitude longitude` , `location`, `position`	Qualsias	CloudWatch I log possono rilevare le coordinate GPS se le coordinate di latitudine e longitudine sono memorizzate come coppia e sono in formato DD (Decimal Degrees), ad esempio 41.948614, -87,655311. Il supporto

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e Regioni	Note
				non include le coordinate nel formato DDM (Gradi Decimali Minuti), ad esempio 41°56.9168'N 87°39.3187'W, o nel formato Gradi, Minuti, Secondi (DMS), ad esempio 41°56'55.0104"N 87°39'19.1196"W.

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e Regioni	Note
Nome completo	`Name`	Nessuno	Qualsias	CloudWatch I log possono rilevare solo i nomi completi. Il supporto è limitato ai set di caratteri latini.

Tipo di dato	ID dell'identificatore di dati	Parola chiave obbligatoria	Paesi e Regioni	Note
Numero di matricola del veicolo (VIN)	`VehicleIdentificationNumber`	`Fahrgestellnummer` , `niv`, `numarul de identificare` , `numarul seriei de sasiu`, `serie sasiu`, `numer VIN`, `Número de Identificação do Veículo`, `Número de Identificación de Automóviles` , `numéro d'identification du véhicule`, `vehicle identification number`, `vin`, `VIN numeris`	Qualsias	CloudWatch I log sono in grado di rilevare VIN costituiti da una sequenza di 17 caratteri e conformi agli standard ISO 3779 e 3780. Questi standard sono stati progettati per l'uso a livello mondiale.

Parole chiave per i numeri identificativi delle patenti di guida

Per rilevare vari tipi di numeri identificativi della patente di guida, CloudWatch Logs richiede che una parola chiave si trovi in prossimità dei numeri. La tabella seguente elenca le parole chiave che CloudWatch Logs riconosce per paesi e aree geografiche specifici.

Paese o regione	Parole chiave
Australia	dl# dl:, dl :, dlno# driver licence, driver license, driver permit, drivers lic., drivers licence, driver's licence, drivers license, driver's license, drivers permit, driver's permit, drivers permit number, driving licence, driving license, driving permit
Austria	führerschein, fuhrerschein, führerschein republik österreich, fuhrerschein republik osterreich
Belgio	fuehrerschein, fuehrerschein- nr, fuehrersc heinnummer, fuhrerschein, führerschein, fuhrerschein- nr, führerschein- nr, fuhrersch einnummer, führerscheinnummer, numéro permis conduire, permis de conduire, rijbewijs, rijbewijsnummer
Bulgaria	превозно средство, свидетелство за управление на моторно, свидетелство за управление на мпс, сумпс, шофьорска книжка
Canada	dl#, dl:, dlno#, driver licence, driver licences, driver license, driver licenses, driver permit, drivers lic., drivers licence, driver's licence, drivers licences, driver's licences, drivers license, driver's license, drivers licenses, driver's licenses, drivers permit, driver's permit,

Paese o regione	Parole chiave
	drivers permit number, driving licence, driving license, driving permit, permis de conduire
Croazia	vozačka dozvola
Cipro	άδεια οδήγησης
Repubblica Ceca	číslo licence, číslo licence řidiče, číslo řidičskéh o průkazu, ovladače lic., povolení k jízdě, povolení řidiče, řidiči povolení, řidičský prúkaz, řidičský průkaz
Danimarca	kørekort, kørekortnummer
Estonia	juhi litsentsi number, juhiloa number, juhiluba, juhiluba number
Finlandia	ajokortin numero, ajokortti, förare lic., körkort, körkort nummer, kuljettaja lic., permis de conduire
Francia	permis de conduire
Germania	fuehrerschein, fuehrerschein- nr, fuehrersc heinnummer, fuhrerschein, führerschein, fuhrerschein- nr, führerschein- nr, fuhrersch einnummer, führerscheinnummer
Grecia	δεια οδήγησης, adeia odigisis
Ungheria	illesztőprogramok lic, jogosítvány, jogsi, licencszám, vezető engedély, vezetői engedély
Irlanda	ceadúnas tiomána
Italia	patente di guida, patente di guida numero, patente guida, patente guida numero

Paese o regione	Parole chiave
Lettonia	autovadītāja apliecība, licences numurs, vadītāja apliecība, vadītāja apliecības numurs, vadītāja atļauja, vadītāja licences numurs, vadītāji lic.
Lituania	vairuotojo pažymėjimas
Lussemburgo	fahrerlaubnis, führerschäin
Malta	liċenzja tas-sewqan
Paesi Bassi	permis de conduire, rijbewijs, rijbewijsnummer
Polonia	numer licencyjny, prawo jazdy, zezwolenie na prowadzenie
Portogallo	carta de condução, carteira de habilitação, carteira de motorist, carteira habilitação, carteira motorist, licença condução, licença de condução, número de licença, número licença, permissão condução, permissão de condução
Romania	numărul permisului de conducere, permis de conducere
Slovacchia	číslo licencie, číslo vodičského preukazu, ovládače lic., povolenia vodičov, povolenie jazdu, povolenie na jazdu, povolenie vodiča, vodičský preukaz
Slovenia	vozniško dovoljenje

Paese o regione	Parole chiave
Spagna	carnet conducer, el carnet de conducer, licencia conducer, licencia de manejo, número carnet conducer, número de carnet de conducer, número de permiso conducer, número de permiso de conducer, número licencia conducer, número permiso conducer, permiso conducción, permiso conducer, permiso de conducción
Svezia	ajokortin numero, dlno# ajokortti, drivere lic., förare lic., körkort, körkort nummer, körkortsn ummer, kuljettajat lic.
Regno Unito	dl#, dl:, dlno#, driver licence, driver licences, driver license, driver licenses, driver permit, drivers lic., drivers licence, driver's licence, drivers licences, driver's licences, drivers license, driver's license, drivers licenses, driver's licenses, drivers permit, driver's permit, drivers permit number, driving licence, driving license, driving permit
Stati Uniti	dl#, dl:, dlno#, driver licence, driver licences, driver license, driver licenses, driver permit, drivers lic., drivers licence, driver's licence, drivers licences, driver's licences, drivers license, driver's license, drivers licenses, driver's licenses, drivers permit, driver's permit, drivers permit number, driving licence, driving license, driving permit

Parole chiave per i numeri di carta d'identità

Per rilevare vari tipi di numeri di identificazione nazionali, CloudWatch Logs richiede che una parola chiave si trovi in prossimità dei numeri. Sono inclusi gli identificatori DNI (Documento Nacional de

Identidad, Spagna), i codici INSEE (Institut national de la statistique et des études économiques), i numeri delle carte d'identità tedesche e i numeri RG (Registro Geral, Brasile).

La tabella seguente elenca le parole chiave che CloudWatch Logs riconosce per paesi e regioni specifici.

Paese o regione	Parole chiave
Brasile	registro geral, rg
Francia	assurance sociale, carte nationale d’identit é, cni, code sécurité sociale, French social security number, fssn#, insee, insurance number, national id number, nationalid#, numéro d'assurance, sécurité sociale, sécurité sociale non., sécurité sociale numéro, social, social security, social security number, socialsecuritynumber, ss#, ssn, ssn#
Germania	ausweisnummer, id number, identification number, identity number, insurance number, personal id, personalausweis
Italia	codice fiscal, dati anagrafici, ehic, health card, health insurance card, p. iva, partita i.v.a., personal data, tax code, tessera sanitaria
Spagna	dni, dni#, dninúmero#, documento nacional de identidad, identidad único, identidadúnico#, insurance number, national identification number, national identity, nationalid#, nationali dno#, número nacional identidad, personal identification number, personal identity no, unique identity number, uniqueid#

Parole chiave per i numeri di passaporto

Per rilevare vari tipi di numeri di passaporto, CloudWatch Logs richiede che una parola chiave si trovi in prossimità dei numeri. La tabella seguente elenca le parole chiave che CloudWatch Logs riconosce per paesi e aree geografiche specifici.

Paese o regione	Parole chiave
Canada	passeport, passeport#, passport, passport#, passportno, passportno#
Francia	numéro de passeport, passeport, passeport #, passeport #, passeportn °, passeport n °, passeportNon, passeport non
Germania	ausstellungsdatum, ausstellungsort, geburtsdatum, passport, passports, reisepass, reisepass–nr, reisepassnummer
Italia	italian passport number, numéro passeport, numéro passeport italien, passaporto, passaporto italiana, passaporto numero, passport number, repubblica italiana passaporto
Spagna	españa pasaporte, libreta pasaporte, número pasaporte, pasaporte, passport, passport book, passport no, passport number, spain passport
Regno Unito	passeport #, passeport n °, passeportNon, passeport non, passeportn °, passport #, passport no, passport number, passport#, passportid
Stati Uniti	passport, travel document

Parole chiave per i numeri di identificazione dei contribuenti e i numeri di riferimento

Per rilevare vari tipi di codice identificativo e di riferimento dei contribuenti, CloudWatch Logs richiede che una parola chiave si trovi in prossimità dei numeri. La tabella seguente elenca le parole chiave che CloudWatch Logs riconosce per paesi e aree geografiche specifici.

Paese o regione	Parole chiave
Brasile	cadastro de pessoa física, cadastro de pessoa fisica, cadastro de pessoas físicas, cadastro de pessoas fisicas, cadastro nacional da pessoa jurídica, cadastro nacional da pessoa juridica, cnpj, cpf
Francia	numéro d'identification fiscale, tax id, tax identification number, tax number, tin, tin#
Germania	identifikationsnummer, steuer id, steueride ntifikationsnummer, steuernummer, tax id, tax identification number, tax number
Spagna	cif, cif número, cifnúmero#, nie, nif, número de contribuyente, número de identidad de extranjero, número de identificación fiscal, número de impuesto corporativo, personal tax number, tax id, tax identification number, tax number, tin, tin#
Regno Unito	paye, tax id, tax id no., tax id number, tax identification, tax identification#, tax no., tax number, tax reference, tax#, taxid#, temporary reference number, tin, trn, unique tax reference, unique taxpayer reference, utr
Stati Uniti	Individual Taxpayer Identification Numbers (ITIN o i.t.i.n.)

ARN degli identificatori dei dati per le informazioni personali di identificazione (PII)

Nella seguente tabella sono elencati i nomi delle risorse Amazon (ARN) per gli identificatori dei dati delle informazioni personali di identificazione (PII) che è possibile aggiungere alle policy di protezione dei dati.

ARN delle informazioni personali di identificazione (PII)
arn:aws:dataprotection::aws:data-identifier/Address
arn:aws:dataprotection::aws:data-identifier/CepCode-BR
arn:aws:dataprotection::aws:data-identifier/Cnpj-BR
arn:aws:dataprotection::aws:data-identifier/CpfCode-BR
arn:aws:dataprotection::aws:data-identifier/DriversLicense-AT
arn:aws:dataprotection::aws:data-identifier/DriversLicense-AU
arn:aws:dataprotection::aws:data-identifier/DriversLicense-BE
arn:aws:dataprotection::aws:data-identifier/DriversLicense-BG
arn:aws:dataprotection::aws:data-identifier/DriversLicense-CA
arn:aws:dataprotection::aws:data-identifier/DriversLicense-CY
arn:aws:dataprotection::aws:data-identifier/DriversLicense-CZ
arn:aws:dataprotection::aws:data-identifier/DriversLicense-DE
arn:aws:dataprotection::aws:data-identifier/DriversLicense-DK
arn:aws:dataprotection::aws:data-identifier/DriversLicense-EE
arn:aws:dataprotection::aws:data-identifier/DriversLicense-ES
arn:aws:dataprotection::aws:data-identifier/DriversLicense-FI
arn:aws:dataprotection::aws:data-identifier/DriversLicense-FR

ARN delle informazioni personali di identificazione (PII)
arn:aws:dataprotection::aws:data-identifier/DriversLicense-GB
arn:aws:dataprotection::aws:data-identifier/DriversLicense-GR
arn:aws:dataprotection::aws:data-identifier/DriversLicense-HR
arn:aws:dataprotection::aws:data-identifier/DriversLicense-HU
arn:aws:dataprotection::aws:data-identifier/DriversLicense-IE
arn:aws:dataprotection::aws:data-identifier/DriversLicense-IT
arn:aws:dataprotection::aws:data-identifier/DriversLicense-LT
arn:aws:dataprotection::aws:data-identifier/DriversLicense-LU
arn:aws:dataprotection::aws:data-identifier/DriversLicense-LV
arn:aws:dataprotection::aws:data-identifier/DriversLicense-MT
arn:aws:dataprotection::aws:data-identifier/DriversLicense-NL
arn:aws:dataprotection::aws:data-identifier/DriversLicense-PL
arn:aws:dataprotection::aws:data-identifier/DriversLicense-PT
arn:aws:dataprotection::aws:data-identifier/DriversLicense-RO
arn:aws:dataprotection::aws:data-identifier/DriversLicense-SE
arn:aws:dataprotection::aws:data-identifier/DriversLicense-SI
arn:aws:dataprotection::aws:data-identifier/DriversLicense-SK
arn:aws:dataprotection::aws:data-identifier/DriversLicense-US
arn:aws:dataprotection::aws:data-identifier/ElectoralRollNumber-GB
arn:aws:dataprotection::aws:data-identifier/EmailAddress

ARN delle informazioni personali di identificazione (PII)
arn:aws:dataprotection::aws:data-identifier/IndividualTaxIdentificationNumber-US
arn:aws:dataprotection::aws:data-identifier/InseeCode-FR
arn:aws:dataprotection::aws:data-identifier/LatLong
arn:aws:dataprotection::aws:data-identifier/Name
arn:aws:dataprotection::aws:data-identifier/NationalIdentificationNumber-DE
arn:aws:dataprotection::aws:data-identifier/NationalIdentificationNumber-ES
arn:aws:dataprotection::aws:data-identifier/NationalIdentificationNumber-IT
arn:aws:dataprotection::aws:data-identifier/NieNumber-ES
arn:aws:dataprotection::aws:data-identifier/NifNumber-ES
arn:aws:dataprotection::aws:data-identifier/PassportNumber-CA
arn:aws:dataprotection::aws:data-identifier/PassportNumber-DE
arn:aws:dataprotection::aws:data-identifier/PassportNumber-ES
arn:aws:dataprotection::aws:data-identifier/PassportNumber-FR
arn:aws:dataprotection::aws:data-identifier/PassportNumber-GB
arn:aws:dataprotection::aws:data-identifier/PassportNumber-IT
arn:aws:dataprotection::aws:data-identifier/PassportNumber-US
arn:aws:dataprotection::aws:data-identifier/PermanentResidenceNumber-CA

ARN delle informazioni personali di identificazione (PII)
arn:aws:dataprotection::aws:data-identifier/PhoneNumber-BR
arn:aws:dataprotection::aws:data-identifier/PhoneNumber-DE
arn:aws:dataprotection::aws:data-identifier/PhoneNumber-ES
arn:aws:dataprotection::aws:data-identifier/PhoneNumber-FR
arn:aws:dataprotection::aws:data-identifier/PhoneNumber-GB
arn:aws:dataprotection::aws:data-identifier/PhoneNumber-IT
arn:aws:dataprotection::aws:data-identifier/PhoneNumber-US
arn:aws:dataprotection::aws:data-identifier/PostalCode-CA
arn:aws:dataprotection::aws:data-identifier/RgNumber-BR
arn:aws:dataprotection::aws:data-identifier/SocialInsuranceNumber-CA
arn:aws:dataprotection::aws:data-identifier/Ssn-ES
arn:aws:dataprotection::aws:data-identifier/Ssn-US
arn:aws:dataprotection::aws:data-identifier/TaxId-DE
arn:aws:dataprotection::aws:data-identifier/TaxId-ES
arn:aws:dataprotection::aws:data-identifier/TaxId-FR
arn:aws:dataprotection::aws:data-identifier/TaxId-GB
arn:aws:dataprotection::aws:data-identifier/VehicleIdentificationNumber
arn:aws:dataprotection::aws:data-identifier/ZipCode-US

Creazione di parametri da log eventi mediante filtri

Puoi cercare e filtrare i dati di registro che entrano in CloudWatch Logs creando uno o più filtri metrici. I filtri metrici definiscono i termini e i modelli da cercare nei dati di registro quando vengono inviati ai registri. CloudWatch CloudWatch Logs utilizza questi filtri metrici per trasformare i dati di registro in CloudWatch metriche numeriche su cui è possibile rappresentare graficamente o impostare un allarme.

Quando si crea un parametro da un filtro di log, è anche possibile scegliere di assegnare dimensioni e un'unità al parametro. Se si sceglie un'unità, assicurarsi di specificare quella corretta quando si crea il filtro. In seguito, la modifica dell'unità per il filtro non avrà alcun effetto.

Puoi utilizzare qualsiasi tipo di CloudWatch statistica, incluse le statistiche sui percentili, quando visualizzi queste metriche o imposti gli allarmi.

> **Note**
>
> Le statistiche dei percentili sono supportate per un parametro solo se nessuno dei valori del parametro è negativo. Se configuri il filtro parametri in modo che possa indicare i numeri negativi, le statistiche dei percentili non saranno disponibili per i parametri quando includono numeri negativi come valore. Per ulteriori informazioni, consulta Percentili.

Filtri non filtrano retroattivamente i dati. I filtri pubblicano solo i punti di dati dei parametri per eventi che accadono dopo la creazione del filtro. I risultati filtrati restituiscono le prime 50 righe, che non verranno visualizzate se il timestamp sui risultati filtrati è antecedente al tempo di creazione dei parametri.

Indice

Concetti

Ogni filtro parametri è formato dai seguenti elementi chiave:

Valore predefinito

Il valore riportato per il filtro di parametri durante un periodo di tempo in cui i log vengono inseriti ma non ne vengono individuati di corrispondenti. Impostandolo su 0, ti assicuri che i dati siano riportati durante ogni periodo, evitando parametri "instabili" con periodi di assenza di dati. Se nessun log viene inserito nell'arco di tempo di un minuto, allora nessun valore viene segnalato.

Se si assegnano dimensioni a un parametro creato da un filtro di parametri, non è possibile assegnare un valore predefinito per tale parametro.

dimensioni

Le dimensioni sono le coppie chiave-valore che definiscono ulteriormente un parametro. È possibile assegnare dimensioni al parametro creato da un filtro di parametri. Poiché le dimensioni fanno parte dell'identificatore univoco di un parametro, ogni volta che una coppia nome/valore unica viene estratta dai log, crei una nuova variante di detto parametro.

Modello di filtro

Una descrizione simbolica di come CloudWatch Logs deve interpretare i dati in ogni evento di registro. Ad esempio, una voce di log potrebbe contenere timestamp, indirizzi IP, stringhe e così via. Utilizza il modello per specificare che cosa ricercare nei file di log.

nome parametro

Il nome della CloudWatch metrica su cui devono essere pubblicate le informazioni di registro monitorate. Ad esempio, puoi pubblicare su una metrica chiamata. ErrorCount

Spazio dei nomi del parametro:

Lo spazio dei nomi di destinazione della nuova metrica. CloudWatch

Valore del parametro:

Il valore numerico da pubblicare nel parametro ogni volta che viene trovato un log corrispondente. Ad esempio, se stai contando le occorrenze di un determinato termine come "Error", il valore sarà "1" per ogni ricorrenza. Se stai contando i byte trasferiti, puoi incrementare con il numero effettivo di byte presenti nel log eventi.

Sintassi del modello di filtro per i filtri di parametri

> **Note**
>
> In che modo i filtri metrici differiscono dalle CloudWatch query di Logs Insights
> I filtri metrici differiscono dalle query di CloudWatch Logs Insights in quanto un valore numerico specificato viene aggiunto a un filtro metrico ogni volta che viene trovato un log corrispondente. Per ulteriori informazioni, consulta Configurazione di valori di parametri per un filtro di parametri.
> Per informazioni su come interrogare i tuoi gruppi di log con il linguaggio di query Amazon CloudWatch Logs Insights, consultaCloudWatch Sintassi delle query di Logs Insights.
> Esempi di modelli di filtro generici
> Per ulteriori informazioni sulla sintassi del modello di filtro generico applicabile ai filtri di parametri, nonché ai filtri di sottoscrizione e ai filtri di log eventi, consulta Sintassi del modello di filtro per filtri di parametri, filtri di sottoscrizione e filtri di log eventi, che include i seguenti esempi:
>
> - Sintassi delle espressioni regolari (regex) supportate
> - Corrispondenza dei termini nei log eventi non strutturati
> - Corrispondenza dei termini negli log eventi JSON
> - Corrispondenza dei termini in log eventi delimitati da spazi

I filtri metrici consentono di cercare e filtrare i dati di log che entrano in CloudWatch Logs, estrarre osservazioni metriche dai dati di log filtrati e trasformare i punti dati in una metrica Logs. CloudWatch Puoi definire i termini e i modelli da cercare nei dati di registro man mano che vengono inviati ai registri. CloudWatch I filtri di parametro vengono assegnati ai gruppi di log e tutti i filtri assegnati a un gruppo di log vengono applicati ai relativi flussi di log.

Quando un modello di filtro trova un termine corrispondente, aumenta il conteggio del parametro di un valore numerico specificato. Ad esempio, puoi creare un filtro di parametri per ricercare e contare l'occorrenza della parola ERROR nei tuoi log eventi.

È possibile assegnare unità e dimensioni ai parametri. Ad esempio, se crei un filtro di parametri che conta l'occorrenza della parola ERROR nei tuoi log eventi, puoi specificare una dimensione chiamata `ErrorCode` per mostrare il numero totale di log eventi che contengono la parola ERROR e filtrare i dati per codici di errore segnalati.

> **Tip**
>
> Quando assegni un'unità di misura a un parametro, assicurati di specificare quella corretta. Se cambi l'unità in un secondo momento, la modifica potrebbe non avere effetto. Per l'elenco completo delle unità CloudWatch supportate, MetricDatumconsulta Amazon CloudWatch API Reference.

Argomenti

Configurazione di valori di parametri per un filtro di parametri

Quando crei un filtro di parametri, definisci il modello di filtro e specifichi il valore del parametro e il valore predefinito. Puoi impostare i valori dei parametri su numeri, identificatori di nome o identificatori numerici. Se non specifichi un valore predefinito, CloudWatch non riporterà i dati quando il filtro metrico non trova una corrispondenza. Consigliamo di specificare un valore predefinito, anche se il valore è 0. L'impostazione di un valore predefinito aiuta a CloudWatch riportare i dati in modo più accurato e CloudWatch impedisce l'aggregazione di metriche disomogenee. CloudWatch aggrega e riporta i valori delle metriche ogni minuto.

Quando il filtro di parametri trova una corrispondenza nei log eventi, aumenta il conteggio del parametro in base al valore dello stesso. Se il filtro metrico non trova una corrispondenza, CloudWatch riporta il valore predefinito della metrica. Ad esempio, il gruppo di log pubblica due record ogni minuto, il valore del parametro è 1 e quello predefinito è 0. Se il filtro di parametri rileva corrispondenze in entrambi i record di log nel primo minuto, il valore del parametro per quel minuto è pari a 2. Se il filtro di parametri non trova corrispondenze in entrambi i record durante il secondo minuto, il valore predefinito per quel minuto è 0. Se assegni dimensioni ai parametri generati dai filtri di parametri, non puoi specificare i valori predefiniti per tali parametri.

Puoi anche impostare un filtro di parametri per incrementare un parametro con un valore estratto da un log eventi, anziché un valore statico. Per ulteriori informazioni, consulta Utilizzo di valori nei log eventi per incrementare il valore di un parametro.

Pubblicazione di dimensioni con parametri da valori in JSON o log eventi delimitati da spazi

Puoi utilizzare la CloudWatch console o la AWS CLI per creare filtri metrici che pubblicano dimensioni con metriche generate da JSON e da eventi di registro delimitati da spazi. Le dimensioni sono coppie nome/valore e sono disponibili solo per modelli di filtro JSON e delimitati da spazi. Puoi creare filtri di parametri JSON e delimitati da spazi con un massimo di tre dimensioni. Per ulteriori informazioni sulle dimensioni e su come assegnare dimensioni ai parametri, consulta le seguenti sezioni:

- Dimensioni nella guida per l' CloudWatch utente di Amazon
- Esempio: estrai campi da un log di Apache e assegna dimensioni nella Amazon CloudWatch Logs User Guide

> **Important**
>
> Le dimensioni contengono valori che raccolgono addebiti identici ai parametri personalizzati. Per evitare addebiti imprevisti, non specificare come dimensioni campi ad alta cardinalità, ad esempio `IPAddress` o `requestID`.
>
> Se estrai parametri dai log eventi, ti verranno fatturati come parametri personalizzati. Per aiutare a prevenire addebiti accidentali elevati, Amazon potrebbe disabilitare un filtro di parametri se genera 1000 coppie nome/valore diverse per le dimensioni specificate entro un determinato periodo di tempo.
>
> Puoi creare allarmi di fatturazione per ricevere una notifica sugli addebiti stimati. Per ulteriori informazioni, consulta Creazione di un allarme di fatturazione per monitorare gli addebiti stimati. AWS

Pubblicazione di dimensioni con parametri dal log eventi JSON

Gli esempi seguenti contengono frammenti di codice che descrivono come specificare le dimensioni in un filtro di parametri JSON.

Example: JSON log event

```
{
  "eventType": "UpdateTrail",
  "sourceIPAddress": "111.111.111.111",
```

```
  "arrayKey": [
        "value",
        "another value"
  ],
  "objectList": [
       {"name": "a",
         "id": 1
       },
       {"name": "b",
         "id": 2
       }
  ]

}
```

> **Note**
>
> Se testi il filtro di parametri di esempio con il log eventi JSON di esempio, devi inserire il log JSON di esempio su una singola riga.

Example: Metric filter

Il filtro di parametri incrementa il parametro ogni volta che un log eventi JSON contiene le proprietà `eventType` e `"sourceIPAddress"`.

```
{ $.eventType = "*" && $.sourceIPAddress != 123.123.* }
```

Quando crei un filtro di parametri JSON, puoi specificare una qualsiasi proprietà del filtro di parametri come dimensione. Ad esempio, per impostare `eventType` come dimensione, segui questa procedura:

```
"eventType" : $.eventType
```

La metrica di esempio contiene una dimensione denominata `"eventType"` e il valore della dimensione nel log eventi di esempio è `"UpdateTrail"`.

Pubblicazione di dimensioni con parametri da log eventi delimitati da spazi

Gli esempi seguenti contengono frammenti di codice che descrivono come specificare le dimensioni in un filtro di parametri delimitato da spazi.

Example: Space-delimited log event

```
127.0.0.1 Prod frank [10/Oct/2000:13:25:15 -0700] "GET /index.html HTTP/1.0" 404
 1534
```

Example: Metric filter

```
[ip, server, username, timestamp, request, status_code, bytes > 1000]
```

Il filtro di parametri incrementa il parametro quando un log eventi delimitato da spazi include uno dei campi specificati nel filtro. Ad esempio, il filtro di parametri trova i seguenti campi e valori nell'esempio del log eventi delimitato da spazi.

```
{
   "$bytes": "1534",
   "$status_code": "404",

   "$request": "GET /index.html HTTP/1.0",
   "$timestamp": "10/Oct/2000:13:25:15 -0700",
   "$username": "frank",
   "$server": "Prod",
   "$ip": "127.0.0.1"
}
```

Quando crei un filtro di parametri delimitato da spazi, puoi specificare uno dei campi nel filtro di parametri come dimensione. Ad esempio, per impostare `server` come dimensione, segui questa procedura:

```
"server" : $server
```

Il filtro di parametri di esempio ha una dimensione denominata `server` e il valore della dimensione nel log eventi di esempio è `"Prod"`.

Example: Match terms with AND (&&) and OR (||)

Puoi utilizzare gli operatori logici AND ("&&") e OR ("||") per creare filtri di parametri delimitati da spazi che contengono condizioni. Il seguente modello di filtro restituisce log eventi in cui la prima parola è ERROR o WARNING.

```
[w1=ERROR || w1=%WARN%, w2]
```

Utilizzo di valori nei log eventi per incrementare il valore di un parametro

Puoi creare filtri di parametri che pubblicano i valori numerici trovati nei log eventi. La procedura in questa sezione utilizza il seguente filtro di parametri di esempio per mostrare come è possibile pubblicare un valore numerico in un log eventi JSON in un parametro.

```
{ $.latency = * } metricValue: $.latency
```

Creazione di un filtro di parametri che pubblica un valore in un log eventi

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione scegli Log, quindi Gruppi di log.
3. Seleziona o crea un gruppo di log.

   Per informazioni su come creare un gruppo di log, consulta Creare un gruppo di log in CloudWatch Logs nella Amazon CloudWatch Logs User Guide.
4. Scegli Actions (Operazioni) e quindi Create metric filter (Crea filtro parametri).
5. In Modello di filtro, digita **`{ $.latency = * }`**, quindi scegli Next (Successivo).
6. In Metric Name (Nome parametro), digita myMetric.
7. Per Valore parametro, immettere **`$.latency`**.
8. (Facoltativo) In Valore predefinito immetti 0, quindi scegli Next (Successivo).

Consigliamo di specificare un valore predefinito, anche se il valore è 0. L'impostazione di un valore predefinito aiuta a CloudWatch riportare i dati in modo più accurato e CloudWatch impedisce l'aggregazione di metriche errate. CloudWatch aggrega e riporta i valori delle metriche ogni minuto.

9. Scegli Crea filtro parametri.

Il filtro di parametri di esempio corrisponde al termine `"latency"` nel log eventi JSON di esempio e pubblica un valore numerico pari a 50 per il parametro myMetric.

```
{
"latency": 50,
"requestType": "GET"
}
```

Creazione di filtri di parametri

La procedura e gli esempi seguenti mostrano come creare filtri di parametri.

Esempi

- Creazione di un filtro di parametri per un gruppo di log
- Esempio: conteggio di log eventi
- Esempio: conteggio delle occorrenze di un termine
- Esempio: conteggio di codici HTTP 404
- Esempio: conteggio di codici HTTP 4xx
- Esempio: estrazione di campi da un log di Apache e assegnare dimensioni

Creazione di un filtro di parametri per un gruppo di log

Per creare un filtro di parametri per un gruppo di log, procedi nel seguente modo. Il parametro non sarà visibile finché non saranno disponibili alcuni punti dati.

Per creare un filtro metrico utilizzando la console CloudWatch

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione scegli Log, quindi Gruppi di log.

3. Scegli il nome del gruppo di log.
4. Scegli `Actions` e quindi Crea filtro parametri.
5. Per Modello di filtro inserisci il modello di filtro. Per ulteriori informazioni, consulta Sintassi del modello di filtro per filtri di parametri, filtri di sottoscrizione, filtri di log eventi e Live Tail.
6. (Facoltativo) Per testare il modello di filtro, in Test Pattern (Modello di test), inserisci uno o più log eventi per testare il modello. Ogni log eventi deve essere formattato su una riga. Le interruzioni di riga vengono utilizzate per separare i log eventi nel riquadro Messaggi del log eventi.
7. Scegli Successivo e poi inserisci un nome per il filtro.
8. In Dettagli della metrica, per Metric namespace, inserisci un nome per lo spazio dei CloudWatch nomi in cui verrà pubblicata la metrica. Se questo spazio dei nomi non esiste già, assicurati che sia selezionato Crea nuovo.
9. Per Metric name (Nome parametro) inserisci un nome per il nuovo parametro.
10. In Metric value (Valore del parametro), se il filtro di parametri conta le occorrenze delle parole chiave nel filtro, inserisci 1. In questo modo il parametro viene incrementato di 1 per ogni log eventi che include una delle parole chiave.

    In alternativa, inserisci un token come **`$size`**. In questo modo il parametro viene incrementato in base al valore del numero nel campo `size` per ogni log eventi che contiene un campo `size`.
11. (Facoltativo) Per Unit (Unità), seleziona un'unità da assegnare al parametro. Se non specifichi un'unità, l'unità viene impostata come `None`.
12. (Facoltativo) Inserisci i nomi e i token per fino a tre dimensioni per il parametro. Se assegni dimensioni ai parametri creati dai filtri di parametri, non puoi assegnare i valori predefiniti per tali parametri.

    > **Note**
    >
    > Le dimensioni sono supportate solo nei filtri JSON o di parametri delimitati da spazi.
13. Scegli Crea filtro parametri. Puoi trovare il filtro dei parametri che hai creato nel riquadro di navigazione. Scegli Log e quindi il gruppo di log. Scegli il nome del gruppo di log per cui hai creato il filtro dei parametri, quindi seleziona il tab Filtri dei parametri.

Esempio: conteggio di log eventi

Il tipo più semplice di monitoraggio di eventi di log è contare il numero di eventi di log che si verificano. Potresti voler eseguire questa operazione per mantenere un conteggio di tutti gli eventi, per creare un monitor in stile "battito cardiaco" o anche solo per provare la creazione di filtri di parametro.

Nel seguente esempio CLI, un filtro metrico chiamato MyAppAccessCount viene applicato al gruppo di log MyApp /access.log per creare la metrica nello spazio dei nomi EventCount . CloudWatch MyNamespace Il filtro è configurato corrispondere a qualsiasi log eventi e per incrementare il parametro con "1".

Per creare un filtro metrico utilizzando la console CloudWatch

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione, selezionare Log groups (Gruppi di log).
3. Scegliere il nome di un gruppo di log.
4. Scegli `Actions`, Crea filtro parametri.
5. Lasciare vuoti i campi Modello di filtro e Seleziona i dati di log per il test .
6. Scegliere Avanti, quindi per Nome filtro, digitare **`EventCount`**.
7. In Metric Details (Dettagli parametro), per Metric Namespace (Spazio dei nomi del parametro), inserisci **`MyNameSpace`**.
8. Per Metric Name (Nome parametro) digita **`MyAppEventCount`**.
9. Verificare che il Valore parametro sia 1. Questo specifica che il conteggio viene incrementato di 1 per ogni log eventi.
10. In Valore predefinito immettere 0, quindi scegliere Avanti. La specificazione di un valore predefinito garantisce che i dati siano riportati anche durante i periodi in cui non si verificano eventi di log, impedendo parametri instabili in cui i dati talvolta non esistono.
11. Scegli Crea filtro parametri.

Per creare un filtro metrico utilizzando il AWS CLI

Al prompt dei comandi, esegui il comando seguente:

```
aws logs put-metric-filter \
  --log-group-name MyApp/access.log \
```

```
  --filter-name EventCount \
  --filter-pattern " " \
  --metric-transformations \
  metricName=MyAppEventCount,metricNamespace=MyNamespace,metricValue=1,defaultValue=0
```

Puoi testare questa nuova policy pubblicando qualsiasi dati di evento. Dovresti vedere i punti dati pubblicati nella metrica MyAppAccessEventCount.

Per pubblicare i dati degli eventi utilizzando il AWS CLI

Al prompt dei comandi, esegui il comando seguente:

```
aws logs put-log-events \
  --log-group-name MyApp/access.log --log-stream-name TestStream1 \
  --log-events \
    timestamp=1394793518000,message="Test event 1" \
    timestamp=1394793518000,message="Test event 2" \
    timestamp=1394793528000,message="This message also contains an Error"
```

Esempio: conteggio delle occorrenze di un termine

Gli eventi di log includono spesso messaggi importanti che desideri contare, forse per quel che concerne il successo o il fallimento delle operazioni. Ad esempio, se una determinata operazione non riesce, potrebbe verificarsi un errore che viene registrato in un file di log. Potresti voler monitorare queste voci per comprendere l'andamento dei tuoi errori.

Nell'esempio sottostante, una filtro parametri viene creato per monitorare il termine Error. La policy è stata creata e aggiunta al gruppo di log /message.log. MyApp CloudWatch Logs pubblica un punto dati nella metrica CloudWatch personalizzata ErrorCount nello spazio dei nomi MyApp/message.log con un valore di «1" per ogni evento contenente Error. Se nessun evento contiene la parola Error, il valore pubblicato è 0. Quando rappresenti graficamente questi dati nella console CloudWatch, assicurati di utilizzare la statistica di somma.

Dopo aver creato un filtro metrico, puoi visualizzare la metrica nella console. CloudWatch Quando si seleziona il parametro da visualizzare, selezionare lo spazio dei nomi del parametro che corrisponde al nome del gruppo di log. Per ulteriori informazioni, consulta Visualizzazione dei parametri disponibili.

Per creare un filtro metrico utilizzando la console CloudWatch

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.

2. Nel pannello di navigazione, selezionare Log groups (Gruppi di log).
3. Scegli il nome del gruppo di log.
4. Scegliere Operazioni, Crea filtro parametri.
5. In Modello di filtro, immettere **`Error`**.

> **Note**
>
> Tutte le voci in Filtra Pattern (Modello File) fanno distinzione tra lettere maiuscole e minuscole.

6. (Facoltativo) Per testare il modello di filtro, in Test Pattern (Modello di test), inserisci uno o più log eventi da utilizzare per testare il modello. Ogni log eventi deve essere all'interno di una riga, in quanto le interruzioni di riga vengono utilizzate per separare i log eventi nella casella Log event messages (Messaggi di registro eventi).
7. Scegliere Avanti, quindi nella pagina Assegna parametro, per Nome filtro, digitare **`MyAppErrorCount`**.
8. In Metric Details (Dettagli parametro), per Metric Namespace (Spazio dei nomi del parametro), inserisci MyNameSpace.
9. Per Metric Name (Nome parametro) digita ErrorCount.
10. Verificare che il Valore parametro sia 1. Questo specifica che il conteggio viene incrementato di 1 per ogni log eventi contente "Error".
11. Per Valore predefinito digitare 0, quindi scegliere Avanti.
12. Scegli Crea filtro parametri.

Per creare un filtro metrico utilizzando il AWS CLI

Al prompt dei comandi, esegui il comando seguente:

```
aws logs put-metric-filter \
  --log-group-name MyApp/message.log \
  --filter-name MyAppErrorCount \
  --filter-pattern 'Error' \
  --metric-transformations \
      metricName=ErrorCount,metricNamespace=MyNamespace,metricValue=1,defaultValue=0
```

Puoi verificare questa nuova policy pubblicando eventi che contengano la parola "Error" nel messaggio.

Per pubblicare eventi utilizzando il AWS CLI

Al prompt dei comandi, esegui il comando seguente. Nota che i modelli fanno distinzione tra lettere maiuscole e minuscole.

```
aws logs put-log-events \
  --log-group-name MyApp/access.log --log-stream-name TestStream1 \
  --log-events \
    timestamp=1394793518000,message="This message contains an Error" \
    timestamp=1394793528000,message="This message also contains an Error"
```

Esempio: conteggio di codici HTTP 404

Utilizzando CloudWatch Logs, è possibile monitorare quante volte i server Apache restituiscono una risposta HTTP 404, che è il codice di risposta per la pagina non trovata. Potresti voler monitorare questo per comprendere quanto spesso i visitatori del tuo sito non sono in grado di individuare le risorse ricercate. Supponiamo che i tuoi record di log siano strutturati per includere le seguenti informazioni per ogni log eventi (visita del sito):

- Indirizzo IP del richiedente
- Identità RFC 1413
- Username
- Timestamp
- Richiedi metodo con risorsa di richiesta e protocollo
- Codice di risposta HTTP per richiesta
- Byte trasferiti nella richiesta

Un esempio del genere potrebbe apparire come segue:

```
127.0.0.1 - frank [10/Oct/2000:13:55:36 -0700] "GET /apache_pb.gif HTTP/1.0" 404 2326
```

Potresti specificare una regola che cerca di confrontare gli eventi di tale struttura per errori HTTP 404, come mostrato nell'esempio seguente:

Per creare un filtro metrico utilizzando la console CloudWatch

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione, selezionare Log groups (Gruppi di log).
3. Scegli `Actions`, Crea filtro parametri.
4. In Modello di filtro, digitare **`[IP, UserInfo, User, Timestamp, RequestInfo, StatusCode=404, Bytes]`**.
5. (Facoltativo) Per testare il modello di filtro, in Test Pattern (Modello di test), inserisci uno o più log eventi da utilizzare per testare il modello. Ogni log eventi deve essere all'interno di una riga, in quanto le interruzioni di riga vengono utilizzate per separare i log eventi nella casella Log event messages (Messaggi di registro eventi).
6. Scegliere Avanti, quindi per Nome filtro, digitare HTTP404Errors.
7. In Dettagli parametro, per Spazio dei nomi del parametro, immettere **`MyNameSpace`**.
8. Per Nome parametro, immettere **`ApacheNotFoundErrorCount`**.
9. Verificare che il Valore parametro sia 1. Questo specifica che il conteggio viene incrementato di 1 per ogni evento di errore 404.
10. In Valore predefinito immettere 0, quindi scegliere Avanti.
11. Scegli Crea filtro parametri.

Per creare un filtro metrico utilizzando il AWS CLI

Al prompt dei comandi, esegui il comando seguente:

```
aws logs put-metric-filter \
  --log-group-name MyApp/access.log \
  --filter-name HTTP404Errors \
  --filter-pattern '[ip, id, user, timestamp, request, status_code=404, size]' \
  --metric-transformations \
      metricName=ApacheNotFoundErrorCount,metricNamespace=MyNamespace,metricValue=1
```

In questo esempio, sono stati utilizzati caratteri letterali, ad esempio le parentesi quadre sinistra e destra, le virgolette e la stringa di caratteri 404. Il modello deve corrispondere con l'intero messaggio del log eventi, in modo da essere considerato per il monitoraggio.

Puoi verificare la creazione del filtro parametri utilizzando il comando describe-metric-filters. L'output visualizzato dovrebbe essere di questo tipo:

```
aws logs describe-metric-filters --log-group-name MyApp/access.log

{
    "metricFilters": [
        {
            "filterName": "HTTP404Errors",
            "metricTransformations": [
                {
                    "metricValue": "1",
                    "metricNamespace": "MyNamespace",
                    "metricName": "ApacheNotFoundErrorCount"
                }
            ],
            "creationTime": 1399277571078,
            "filterPattern": "[ip, id, user, timestamp, request, status_code=404,
 size]"
        }
    ]
}
```

Ora puoi pubblicare alcuni eventi manualmente:

```
aws logs put-log-events \
--log-group-name MyApp/access.log --log-stream-name hostname \
--log-events \
timestamp=1394793518000,message="127.0.0.1 - bob [10/Oct/2000:13:55:36 -0700] \"GET /
apache_pb.gif HTTP/1.0\" 404 2326" \
timestamp=1394793528000,message="127.0.0.1 - bob [10/Oct/2000:13:55:36 -0700] \"GET /
apache_pb2.gif HTTP/1.0\" 200 2326"
```

Subito dopo aver inserito questi eventi di registro di esempio, puoi recuperare la metrica denominata nella CloudWatch console come. ApacheNotFoundErrorCount

Esempio: conteggio di codici HTTP 4xx

Come nell'esempio precedente, potresti voler monitorare i tuoi log di accesso al servizio Web e monitorare i livelli del codice di risposta HTTP. Ad esempio, potresti voler monitorare tutti gli errori di livello HTTP 400. Tuttavia, è possibile tu non voglia specificare un nuovo filtro parametri per ogni codice restituito.

L'esempio seguente spiega come creare un parametro che includa tutte le risposte del codice di livello HTTP 400 da un log di accesso utilizzando il formato di log di accesso Apache dall'esempio Esempio: conteggio di codici HTTP 404.

Per creare un filtro metrico utilizzando la console CloudWatch

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione, selezionare Log groups (Gruppi di log).
3. Scegli il nome del gruppo di log per il server Apache.
4. Scegli `Actions`, Crea filtro parametri.
5. In Filter pattern (Modello di filtro), inserisci **`[ip, id, user, timestamp, request, status_code=4*, size]`**.
6. (Facoltativo) Per testare il modello di filtro, in Modello di test, inserisci uno o più log eventi da utilizzare per testare il modello. Ogni log eventi deve essere all'interno di una riga, in quanto le interruzioni di riga vengono utilizzate per separare i log eventi nella casella Log event messages (Messaggi di registro eventi).
7. Scegli Next (Avanti), quindi per Filter name (Nome filtro), digita **`HTTP4xxErrors`**.
8. In Metric details (Dettagli parametro), per Metric namespace (Spazio dei nomi del parametro), inserisci **`MyNameSpace`**.
9. Per Metric name (Nome parametro), inserisci HTTP4xxErrors.
10. In Metric value (Valore parametro), inserisci 1. Questo specifica che il conteggio viene incrementato di 1 per ogni log contente un errore 4xx.
11. In Default value (Valore predefinito) inserisci 0, quindi scegli Next (Avanti).
12. Scegli Crea filtro parametri.

Per creare un filtro metrico utilizzando il AWS CLI

Al prompt dei comandi, esegui il comando seguente:

```
aws logs put-metric-filter \
  --log-group-name MyApp/access.log \
  --filter-name HTTP4xxErrors \
  --filter-pattern '[ip, id, user, timestamp, request, status_code=4*, size]' \
  --metric-transformations \
  metricName=HTTP4xxErrors,metricNamespace=MyNamespace,metricValue=1,defaultValue=0
```

Puoi utilizzare i seguenti dati in chiamate put-event per verificare questa regola. Se non rimuovi la regola di monitoraggio nell'esempio precedente, genererai due parametri differenti.

```
127.0.0.1 - - [24/Sep/2013:11:49:52 -0700] "GET /index.html HTTP/1.1" 404 287
127.0.0.1 - - [24/Sep/2013:11:49:52 -0700] "GET /index.html HTTP/1.1" 404 287
127.0.0.1 - - [24/Sep/2013:11:50:51 -0700] "GET /~test/ HTTP/1.1" 200 3
127.0.0.1 - - [24/Sep/2013:11:50:51 -0700] "GET /favicon.ico HTTP/1.1" 404 308
127.0.0.1 - - [24/Sep/2013:11:50:51 -0700] "GET /favicon.ico HTTP/1.1" 404 308
127.0.0.1 - - [24/Sep/2013:11:51:34 -0700] "GET /~test/index.html HTTP/1.1" 200 3
```

Esempio: estrazione di campi da un log di Apache e assegnare dimensioni

A volte, invece di contare, è utile utilizzare i valori all'interno dei singoli eventi di log per i valori di parametro. Questo esempio illustra il modo in cui puoi creare una regola di estrazione per creare un parametro che misura i byte trasferiti da un server Web Apache.

Questa regola di estrazione corrisponde ai sette campi del log eventi. Il valore del parametro è il valore del settimo token corrispondente. Puoi visualizzare il riferimento al token come "$7" nel campo `metricValue` della regola di estrazione.

In questo esempio viene inoltre illustrato come assegnare le dimensioni al parametro che si sta creando.

Per creare un filtro metrico utilizzando la console CloudWatch

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione, selezionare Log groups (Gruppi di log).
3. Scegli il nome del gruppo di log per il server Apache.
4. Scegli `Actions`, Crea filtro parametri.
5. In Filter pattern (Modello di filtro), inserisci **`[ip, id, user, timestamp, request, status_code, size]`**.
6. (Facoltativo) Per testare il modello di filtro, in Modello di test, inserisci uno o più log eventi da utilizzare per testare il modello. Ogni log eventi deve essere all'interno di una riga, in quanto le interruzioni di riga vengono utilizzate per separare i log eventi nella casella Log event messages (Messaggi di registro eventi).
7. Scegli Next (Avanti), quindi per Filter name (Nome filtro), digita **`size`**.

8. In Metric details (Dettagli parametro), per Metric namespace (Spazio dei nomi del parametro), inserisci **`MyNameSpace`**. Poiché si tratta di un nuovo spazio dei nomi, assicurarsi che sia selezionato Create new (Creazione di nuovi).
9. Per Metric name (Nome parametro), inserisci **`BytesTransferred`**
10. In Metric value (Valore parametro), inserisci **`$size`**.
11. Per Unit (Unità) seleziona Bytes (Byte).
12. In Dimension Name (Nome dimensione) digita **`IP`**.
13. Per Dimension Value (Valore dimensione) digita **`$ip`**, quindi scegli Next (Avanti).
14. Scegli Crea filtro parametri.

Per creare questo filtro metrico utilizzando il AWS CLI

Al prompt dei comandi, esegui il comando seguente.

```
aws logs put-metric-filter \
--log-group-name MyApp/access.log \
 --filter-name BytesTransferred \
 --filter-pattern '[ip, id, user, timestamp, request, status_code, size]' \
 --metric-transformations  \
 metricName=BytesTransferred,metricNamespace=MyNamespace,metricValue='$size'
```

```
aws logs put-metric-filter \
--log-group-name MyApp/access.log \
--filter-name BytesTransferred \
--filter-pattern '[ip, id, user, timestamp, request, status_code, size]' \
--metric-transformations  \
metricName=BytesTransferred,metricNamespace=MyNamespace,metricValue='$size',unit=Bytes,dimensio
$ip}}'
```

> **Note**
>
> In questo comando, utilizzare questo formato per specificare più dimensioni.
>
> ```
> aws logs put-metric-filter \
> --log-group-name my-log-group-name \
> --filter-name my-filter-name \
> --filter-pattern 'my-filter-pattern' \
> --metric-transformations  \
> ```

```
metricName=my-metric-name,metricNamespace=my-metric-namespace,metricValue=my-token,unit=unit,dimensions='{dimension1=$dim,dimension2=$dim2,dim3=$dim3}'
```

È possibile utilizzare i seguenti dati nelle put-log-event chiamate per testare questa regola. Questo genererà due parametri differenti, se non rimuovi la regola di monitoraggio nell'esempio precedente.

```
127.0.0.1 - - [24/Sep/2013:11:49:52 -0700] "GET /index.html HTTP/1.1" 404 287
127.0.0.1 - - [24/Sep/2013:11:49:52 -0700] "GET /index.html HTTP/1.1" 404 287
127.0.0.1 - - [24/Sep/2013:11:50:51 -0700] "GET /~test/ HTTP/1.1" 200 3
127.0.0.1 - - [24/Sep/2013:11:50:51 -0700] "GET /favicon.ico HTTP/1.1" 404 308
127.0.0.1 - - [24/Sep/2013:11:50:51 -0700] "GET /favicon.ico HTTP/1.1" 404 308
127.0.0.1 - - [24/Sep/2013:11:51:34 -0700] "GET /~test/index.html HTTP/1.1" 200 3
```

Elencazione di filtri di parametri

Puoi creare un elenco di tutti i filtri di parametri in un gruppo di log.

Per elencare i filtri metrici utilizzando la console CloudWatch

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione, selezionare Log groups (Gruppi di log).
3. Nel riquadro contenuti, nell'elenco dei gruppi di log, nella colonna Filtri di parametri, seleziona il numero di filtri.

   Nella schermata Gruppi di log > Filtri per sono elencati tutti i filtri di parametri associati al gruppo di log.

Per elencare i filtri metrici utilizzando il AWS CLI

Al prompt dei comandi, esegui il comando seguente:

```
aws logs describe-metric-filters --log-group-name MyApp/access.log
```

Di seguito è riportato un output di esempio:

```
{
    "metricFilters": [
```

```
        {
            "filterName": "HTTP404Errors",
            "metricTransformations": [
                {
                    "metricValue": "1",
                    "metricNamespace": "MyNamespace",
                    "metricName": "ApacheNotFoundErrorCount"
                }
            ],
            "creationTime": 1399277571078,
            "filterPattern": "[ip, id, user, timestamp, request, status_code=404,
 size]"
        }
    ]
}
```

Eliminazione di un filtro di parametri

Una policy è identificata dal relativo nome e dal gruppo di log a cui appartiene.

Per eliminare un filtro metrico utilizzando la console CloudWatch

1. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione, selezionare Log groups (Gruppi di log).
3. Nel riquadro contenuti, nella colonna Metric Filters (Filtro di parametri), seleziona il numero di filtri di parametri per il gruppo di log.
4. Nella schermata Metric Filters (Filtri di parametri) seleziona la casella corrispondente al nome del filtro che si desidera eliminare. Scegli Elimina.
5. Quando viene richiesta la conferma, seleziona Delete (Elimina).

Per eliminare un filtro metrico utilizzando il AWS CLI

Al prompt dei comandi, esegui il comando seguente:

```
aws logs delete-metric-filter --log-group-name MyApp/access.log \
 --filter-name MyFilterName
```

Elaborazione in tempo reale dei dati di log con le sottoscrizioni

Puoi utilizzare gli abbonamenti per accedere a un feed di eventi di log in tempo reale da CloudWatch Logs e distribuirlo ad altri servizi come uno stream Amazon Kinesis, uno stream Amazon Kinesis Data Firehose o per l'elaborazione AWS Lambda , l'analisi o il caricamento personalizzati su altri sistemi. Quando i log eventi vengono inviati al servizio ricevente, vengono codificati in formato base64 e compressi nel formato gzip.

Per iniziare la sottoscrizione a log eventi, crea la risorsa ricevente, ad esempio un flusso Kinesis Data Streams, in cui verranno distribuiti gli eventi. Un filtro di sottoscrizione definisce lo schema di filtro da utilizzare per filtrare gli eventi di registro che vengono recapitati alla tua AWS risorsa, oltre a informazioni su dove inviare gli eventi di log corrispondenti.

A ogni gruppo di log può essere associati fino a due filtri di sottoscrizione.

> **Note**
>
> Se il servizio di destinazione restituisce un errore riutilizzabile, ad esempio un'eccezione di limitazione o un'eccezione del servizio riutilizzabile (ad esempio HTTP 5xx), CloudWatch Logs continua a ritentare l'invio per un massimo di 24 ore. CloudWatch Logs non tenta di ripetere la consegna se l'errore è un errore irreversibile, ad esempio o. AccessDeniedException ResourceNotFoundException

CloudWatch Logs produce anche CloudWatch metriche sull'inoltro degli eventi di registro alle sottoscrizioni. Per ulteriori informazioni, consulta Monitoraggio dell'utilizzo con i parametri di CloudWatch.

Puoi anche utilizzare un abbonamento CloudWatch Logs per trasmettere i dati di log quasi in tempo reale a un cluster Amazon OpenSearch Service. Per ulteriori informazioni, consulta Streaming CloudWatch Logs data to Amazon OpenSearch Service.

Indice

- Prevenzione del "confused deputy"

Concetti

Ogni filtro sottoscrizione è formato dai seguenti elementi chiave.

nome gruppo di log

Il gruppo di log a cui associare il filtro sottoscrizione. Tutti i log eventi caricati a questo gruppo di log sono soggetti al filtro di sottoscrizione e quelli che corrispondono al filtro vengono consegnati al servizio di destinazione che riceve i log eventi corrispondenti.

Modello di filtro

Una descrizione simbolica di come CloudWatch Logs dovrebbe interpretare i dati in ogni evento di registro, insieme a espressioni di filtro che limitano ciò che viene consegnato alla risorsa di destinazione. AWS Per ulteriori informazioni sulla sintassi del modello di filtro, consulta Sintassi del modello di filtro per filtri di parametri, filtri di sottoscrizione, filtri di log eventi e Live Tail.

Arn di destinazione

Il nome della risorsa Amazon (ARN) del flusso Kinesis Data Streams, del flusso Kinesis Data Firehose o della funzione Lambda che vuoi utilizzare come destinazione del feed di sottoscrizione.

Arn del ruolo

Un ruolo IAM che concede a CloudWatch Logs le autorizzazioni necessarie per inserire i dati nella destinazione scelta. Questo ruolo non è necessario per le destinazioni Lambda perché CloudWatch i log possono ottenere le autorizzazioni necessarie dalle impostazioni di controllo degli accessi sulla funzione Lambda stessa.

distribuzione

Il metodo utilizzato per distribuire i dati di log alla destinazione, quando la destinazione è un flusso di dati Amazon Kinesis. Per impostazione predefinita, i dati di log vengono raggruppati dal flusso di log. Per una distribuzione più uniforme, puoi raggruppare i dati di log casualmente.

Utilizzo dei filtri di abbonamento Logs CloudWatch

È possibile utilizzare un filtro di sottoscrizione con Kinesis Data Streams, Lambda o Kinesis Data Firehose. I log inviati a un servizio ricevente tramite un filtro di sottoscrizione sono codificati in formato base64 e compressi nel formato gzip.

È possibile eseguire ricerche nei dati di log utilizzando la sintassi filtro e modello.

Esempi

- Esempio 1: filtri di sottoscrizione con Kinesis Data Streams
- Esempio 2: filtri di abbonamento con AWS Lambda
- Esempio 3: filtri di sottoscrizione con Amazon Kinesis Data Firehose

Esempio 1: filtri di sottoscrizione con Kinesis Data Streams

L'esempio seguente associa un filtro di sottoscrizione a un gruppo di log contenente AWS CloudTrail eventi. Il filtro di sottoscrizione invia ogni attività registrata effettuata dalle AWS credenziali «Root» a un flusso in Kinesis Data Streams chiamato "». RootAccess Per ulteriori informazioni su come inviare AWS CloudTrail eventi ai registri, consulta Invio CloudTrail di eventi ai CloudWatch registri nella Guida per CloudWatch l'utente.AWS CloudTrail

> **Note**
>
> Prima di creare il flusso , calcola il volume dei dati di log che verrà generato. Assicurati di creare un flusso che includa partizioni sufficienti per gestire questo volume. Se il flusso non dispone di shard sufficienti, il flusso di log verrà limitato. Per ulteriori informazioni sui limiti dei volumi di flussi, consulta Quote e limiti.
> Vengono fatti nuovi tentativi per i risultati limitati per un massimo di 24 ore. Dopo 24 ore, i risultati non riusciti saranno eliminati.
> Per ridurre il rischio di limitazione, procedi nel seguente modo:
>
> - Monitora il tuo streaming utilizzando CloudWatch le metriche. Questo ti aiuta a identificare eventuali limitazioni e a regolare la configurazione di conseguenza. Ad esempio, la `DeliveryThrottling` metrica può essere utilizzata per tenere traccia del numero di eventi di registro per i quali CloudWatch Logs è stato limitato durante l'inoltro dei dati alla destinazione dell'abbonamento. Per ulteriori informazioni sul monitoraggio, consulta Monitoraggio dell'utilizzo con i parametri di CloudWatch.
> - Usa la modalità di capacità on demand per il tuo flusso in Kinesis Data Streams. La modalità on demand si adatta istantaneamente ai carichi di lavoro siano essi aumentati o diminuiti. Per ulteriori informazioni sulla modalità di capacità on demand, consulta Modalità on demand.

- Limita il modello di filtro dell' CloudWatch abbonamento in modo che corrisponda alla capacità del tuo stream in Kinesis Data Streams. Se invii una quantità eccessiva di dati al flusso, potrebbe essere necessario ridurre le dimensioni del filtro o modificarne i criteri.

Creare un filtro di sottoscrizione per Kinesis Data Streams

1. Crea un flusso di destinazione utilizzando il comando seguente:

```
$ C:\>  aws kinesis create-stream --stream-name "RootAccess" --shard-count 1
```

2. Attendi finché il flusso non diventa attivo. Questa operazione può richiedere uno o due minuti. È possibile utilizzare il seguente comando Kinesis Data Streams describe-stream per controllare il. StreamDescription StreamStatusproprietà. Inoltre, prendi nota del valore StreamDescription.StreamArn, poiché ti servirà in un passaggio successivo:

```
aws kinesis describe-stream --stream-name "RootAccess"
```

Di seguito è riportato un output di esempio:

```
{
    "StreamDescription": {
        "StreamStatus": "ACTIVE",
        "StreamName": "RootAccess",
        "StreamARN": "arn:aws:kinesis:us-east-1:123456789012:stream/RootAccess",
        "Shards": [
            {
                "ShardId": "shardId-000000000000",
                "HashKeyRange": {
                    "EndingHashKey": "340282366920938463463374607431768211455",
                    "StartingHashKey": "0"
                },
                "SequenceNumberRange": {
                    "StartingSequenceNumber":
                    "49551135218688818456679503831981458784591352702181572610"
                }
            }
        ]
    }
}
```

3. Crea il ruolo IAM che concederà a CloudWatch Logs l'autorizzazione a inserire dati nel tuo stream. In primo luogo, sarà necessario creare una policy di attendibilità in un file (ad esempio, `~/TrustPolicyForCWL-Kinesis.json`). Utilizza un editor di testo per creare questa policy. Non utilizzare la console IAM per crearla.

   Questa policy include una chiave di contesto della condizione globale `aws:SourceArn` per prevenire il problema di sicurezza noto come "confused deputy". Per ulteriori informazioni, consulta Prevenzione del "confused deputy".

   ```
   {
     "Statement": {
       "Effect": "Allow",
       "Principal": { "Service": "logs.amazonaws.com" },
       "Action": "sts:AssumeRole",
       "Condition": {
           "StringLike": { "aws:SourceArn": "arn:aws:logs:region:123456789012:*" }
        }
      }
   }
   ```

4. Utilizza il comando create-role per creare un ruolo IAM, specificando il file della policy di attendibilità. Annota il valore Role.Arn restituito, poiché ne avrai bisogno anche in una fase successiva:

   ```
   aws iam create-role --role-name CWLtoKinesisRole --assume-role-policy-document
    file://~/TrustPolicyForCWL-Kinesis.json
   ```

   Di seguito è riportato un esempio di output.

   ```
   {
       "Role": {
           "AssumeRolePolicyDocument": {
               "Statement": {
                   "Action": "sts:AssumeRole",
                   "Effect": "Allow",
                   "Principal": {
                       "Service": "logs.amazonaws.com"
                   },
                   "Condition": {
                       "StringLike": {
                           "aws:SourceArn": { "arn:aws:logs:region:123456789012:*" }
   ```

```
                    }
                }
            }
        },
        "RoleId": "AAOIIAH450GAB4HC5F431",
        "CreateDate": "2015-05-29T13:46:29.431Z",
        "RoleName": "CWLtoKinesisRole",
        "Path": "/",
        "Arn": "arn:aws:iam::123456789012:role/CWLtoKinesisRole"
    }
}
```

5. Crea una politica di autorizzazioni per definire quali azioni CloudWatch Logs può eseguire sul tuo account. In primo luogo, creerai una policy delle autorizzazioni in un file (ad esempio, `~/PermissionsForCWL-Kinesis.json`). Utilizza un editor di testo per creare questa policy. Non utilizzare la console IAM per crearla.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "kinesis:PutRecord",
      "Resource": "arn:aws:kinesis:region:123456789012:stream/RootAccess"
    }
  ]
}
```

6. Associa la politica delle autorizzazioni al ruolo utilizzando il seguente comando: put-role-policy

```
aws iam put-role-policy  --role-name CWLtoKinesisRole  --policy-name Permissions-Policy-For-CWL  --policy-document file://~/PermissionsForCWL-Kinesis.json
```

7. Dopo che lo stream è nello stato Attivo e hai creato il ruolo IAM, puoi creare il filtro di sottoscrizione CloudWatch Logs. Il filtro sottoscrizioni avvia immediatamente il flusso di dati di log in tempo reale dal gruppo di log selezionato nel flusso :

```
aws logs put-subscription-filter \
    --log-group-name "CloudTrail/logs" \
    --filter-name "RootAccess" \
    --filter-pattern "{$.userIdentity.type = Root}" \
    --destination-arn "arn:aws:kinesis:region:123456789012:stream/RootAccess" \
    --role-arn "arn:aws:iam::123456789012:role/CWLtoKinesisRole"
```

8. Dopo aver impostato il filtro di abbonamento, CloudWatch Logs inoltra tutti gli eventi di registro in entrata che corrispondono al modello di filtro allo stream. Puoi verificare che questo stia effettivamente avvenendo acquisendo un'iteratore di partizione di Kinesis Data Streams e utilizzando il comando Kinesis get-records per recuperare alcuni log di Kinesis Data Streams:

```
aws kinesis get-shard-iterator --stream-name RootAccess --shard-id
 shardId-000000000000 --shard-iterator-type TRIM_HORIZON
```

```
{
    "ShardIterator":
    "AAAAAAAAAAFGU/
kLvNggvndHq2UIFOw5PZc6F01s3e3afsSscRM70JSbjIefg2ub07nk1y6CDxYR1UoGHJNP4m4NFUetzfL
+wev+e2P4djJg4L9wmXKvQYoE+rMUiFq
+p4Cn3IgvqOb5dRA0yybNdRcdzvnC35KQANoHzzahKdRGb9v4scv+3vaq+f+OIK8zM5My8ID
+g6rMo7UKWeI4+IWiK2OSh0uP"
}
```

```
aws kinesis get-records --limit 10 --shard-iterator "AAAAAAAAAAFGU/
kLvNggvndHq2UIFOw5PZc6F01s3e3afsSscRM70JSbjIefg2ub07nk1y6CDxYR1UoGHJNP4m4NFUetzfL
+wev+e2P4djJg4L9wmXKvQYoE+rMUiFq
+p4Cn3IgvqOb5dRA0yybNdRcdzvnC35KQANoHzzahKdRGb9v4scv+3vaq+f+OIK8zM5My8ID
+g6rMo7UKWeI4+IWiK2OSh0uP"
```

Potresti dover effettuare questa chiamata più volte prima che Kinesis Data Streams inizi a restituire dati.

In genere si visualizzerà una risposta con una matrice di record. L'attributo Dati in un record Kinesis Data Streams usa la codifica base64 e la compressione nel formato gzip. Puoi esaminare i dati non elaborati dalla riga di comando utilizzando i seguenti comandi Unix:

```
echo -n "<Content of Data>" | base64 -d | zcat
```

I dati con codifica base64 e decompressi sono in formato JSON con la seguente struttura:

```
{
    "owner": "111111111111",
    "logGroup": "CloudTrail/logs",
    "logStream": "111111111111_CloudTrail/logs_us-east-1",
    "subscriptionFilters": [
```

```
        "Destination"
    ],
    "messageType": "DATA_MESSAGE",
    "logEvents": [
        {
            "id": "31953106606966983378809025079804211143289615424298221568",
            "timestamp": 1432826855000,
            "message": "{\"eventVersion\":\"1.03\",\"userIdentity\":{\"type\":
\"Root\"}"
        },
        {
            "id": "31953106606966983378809025079804211143289615424298221569",
            "timestamp": 1432826855000,
            "message": "{\"eventVersion\":\"1.03\",\"userIdentity\":{\"type\":
\"Root\"}"
        },
        {
            "id": "31953106606966983378809025079804211143289615424298221570",
            "timestamp": 1432826855000,
            "message": "{\"eventVersion\":\"1.03\",\"userIdentity\":{\"type\":
\"Root\"}"
        }
    ]
}
```

Gli elementi chiave nella struttura di dati precedente sono i seguenti:

owner

L'ID dell' AWS account dei dati di registro di origine.

logGroup

Nome del gruppo di log dei dati di log originari.

logStream

Nome del flusso di log dei dati di log originari.

subscriptionFilters

Elenco dei nomi di filtro sottoscrizione che corrispondono con i dati di log originari.

messageType

I messaggi di dati utilizzeranno il tipo "DATA_MESSAGE". A volte CloudWatch i log possono emettere record Kinesis Data Streams di tipo «CONTROL_MESSAGE», principalmente per verificare se la destinazione è raggiungibile.

logEvents

I dati di log effettivi, rappresentati come una varietà di record di eventi di log. La proprietà "id" è un identificatore univoco per ogni log eventi.

Esempio 2: filtri di abbonamento con AWS Lambda

In questo esempio, creerai un filtro di sottoscrizione CloudWatch Logs che invia i dati di registro alla tua AWS Lambda funzione.

> **Note**
>
> Prima di creare la funzione Lambda, calcola il volume dei dati di log che verrà generato. Assicurati di creare una funzione che possa gestire questo volume. Se la funzione non dispone di volume sufficiente, il flusso di log verrà limitato. Per ulteriori informazioni sui limiti di Lambda, consulta Limiti diAWS Lambda.

Creazione di un filtro di sottoscrizione per Lambda

1. Crea la AWS Lambda funzione.

   Assicurati di aver configurato il ruolo di esecuzione di Lambda. Per ulteriori informazioni, consulta Fase 2.2: creazione di un ruolo IAM (ruolo di esecuzione) nella Guida per gli sviluppatoriAWS Lambda .
2. Apri un editor di testo e crea un file denominato `helloWorld.js` con il seguente contenuto:

```
var zlib = require('zlib');
exports.handler = function(input, context) {
    var payload = Buffer.from(input.awslogs.data, 'base64');
    zlib.gunzip(payload, function(e, result) {
        if (e) {
            context.fail(e);
        } else {
```

```
            result = JSON.parse(result.toString());
            console.log("Event Data:", JSON.stringify(result, null, 2));
            context.succeed();
        }
    });
};
```

3. Comprimi il file helloWorld.js e salvalo con il nome `helloWorld.zip`.
4. Utilizza il comando seguente, in cui il ruolo è il ruolo di esecuzione Lambda configurato nella prima fase:

```
aws lambda create-function \
    --function-name helloworld \
    --zip-file fileb://file-path/helloWorld.zip \
    --role lambda-execution-role-arn \
    --handler helloWorld.handler \
    --runtime nodejs12.x
```

5. Concedi a CloudWatch Logs il permesso di eseguire la tua funzione. Utilizza il comando seguente, sostituendo l'account di segnaposto con il tuo account e il gruppo di log di segnaposto con il gruppo di log da elaborare:

```
aws lambda add-permission \
    --function-name "helloworld" \
    --statement-id "helloworld" \
    --principal "logs.amazonaws.com" \
    --action "lambda:InvokeFunction" \
    --source-arn "arn:aws:logs:region:123456789123:log-group:TestLambda:*" \
    --source-account "123456789012"
```

6. Crea un filtro di sottoscrizione utilizzando il comando seguente, sostituendo l'account di segnaposto con il tuo account e il gruppo di log di segnaposto con il gruppo di log da elaborare:

```
aws logs put-subscription-filter \
    --log-group-name myLogGroup \
    --filter-name demo \
    --filter-pattern "" \
    --destination-arn arn:aws:lambda:region:123456789123:function:helloworld
```

7. Verifica tramite un log eventi di esempio (facoltativo). Al prompt dei comandi, esegui il comando seguente, il quale inserirà un semplice messaggio di log nel flusso sottoscritto.

Per visualizzare l'output della funzione Lambda, passa alla funzione Lambda, in cui potrai visualizzare l'output in /aws/lambda/helloworld:

```
aws logs put-log-events --log-group-name myLogGroup --log-stream-name stream1 --
log-events "[{\"timestamp\":<CURRENT TIMESTAMP MILLIS> , \"message\": \"Simple
 Lambda Test\"}]"
```

Dovresti visualizzare una risposta con una matrice di Lambda. L'attributo Data (Dati) nel log Lambda usa la codifica base64 e la compressione nel formato gzip. Il payload effettivo che Lambda riceve è nel formato seguente, `{ "awslogs": {"data": "BASE64ENCODED_GZIP_COMPRESSED_DATA"} }`. Puoi esaminare i dati non elaborati dalla riga di comando utilizzando i seguenti comandi Unix:

```
echo -n "<BASE64ENCODED_GZIP_COMPRESSED_DATA>" | base64 -d | zcat
```

I dati con codifica base64 e decompressi sono in formato JSON con la seguente struttura:

```
{
    "owner": "123456789012",
    "logGroup": "CloudTrail",
    "logStream": "123456789012_CloudTrail_us-east-1",
    "subscriptionFilters": [
        "Destination"
    ],
    "messageType": "DATA_MESSAGE",
    "logEvents": [
        {
            "id": "31953106606966983378809025079804211143289615424298221568",
            "timestamp": 1432826855000,
            "message": "{\"eventVersion\":\"1.03\",\"userIdentity\":{\"type\":
\"Root\"}"
        },
        {
            "id": "31953106606966983378809025079804211143289615424298221569",
            "timestamp": 1432826855000,
            "message": "{\"eventVersion\":\"1.03\",\"userIdentity\":{\"type\":
\"Root\"}"
        },
        {
            "id": "31953106606966983378809025079804211143289615424298221570",
```

```
            "timestamp": 1432826855000,
            "message": "{\"eventVersion\":\"1.03\",\"userIdentity\":{\"type\":
\"Root\"}"
        }
    ]
}
```

Gli elementi chiave nella struttura di dati precedente sono i seguenti:

owner

L'ID AWS dell'account dei dati di registro di origine.

logGroup

Nome del gruppo di log dei dati di log originari.

logStream

Nome del flusso di log dei dati di log originari.

subscriptionFilters

Elenco dei nomi di filtro sottoscrizione che corrispondono con i dati di log originari.

messageType

I messaggi di dati utilizzeranno il tipo "DATA_MESSAGE". A volte CloudWatch i log possono emettere record Lambda di tipo «CONTROL_MESSAGE», principalmente per verificare se la destinazione è raggiungibile.

logEvents

I dati di log effettivi, rappresentati come una varietà di record di eventi di log. La proprietà "id" è un identificatore univoco per ogni log eventi.

Esempio 3: filtri di sottoscrizione con Amazon Kinesis Data Firehose

In questo esempio, creerai un abbonamento CloudWatch Logs che invia tutti gli eventi di log in entrata che corrispondono ai filtri definiti al flusso di distribuzione di Amazon Kinesis Data Firehose. I dati inviati dai CloudWatch log ad Amazon Kinesis Data Firehose sono già compressi con la compressione gzip di livello 6, quindi non è necessario utilizzare la compressione all'interno del flusso di distribuzione di Kinesis Data Firehose.

> **Note**
>
> Prima di creare il flusso Kinesis Data Firehose, calcola il volume dei dati di log che verrà generato. Assicurati di creare un flusso Kinesis Data Firehose che possa gestire questo volume. Se il flusso non è in grado di gestire il volume, il flusso di log verrà limitato. Per ulteriori informazioni sui limiti dei volumi di flussi di Kinesis Data Firehose, consulta Limiti dei dati Amazon Kinesis Data Firehose.

Per creare un filtro di sottoscrizione per Kinesis Data Firehose

1. Crea un bucket Amazon Simple Storage Service (Amazon S3). Ti consigliamo di utilizzare un bucket creato appositamente per Logs. CloudWatch Tuttavia, se intendi utilizzare un bucket esistente, puoi passare alla fase 2.

   Esegui il comando seguente, sostituendo il segnaposto Regione con la Regione che desideri utilizzare:

   ```
   aws s3api create-bucket --bucket my-bucket --create-bucket-configuration
    LocationConstraint=region
   ```

   Di seguito è riportato un output di esempio:

   ```
   {
       "Location": "/my-bucket"
   }
   ```

2. Crea il ruolo IAM che concede ad Amazon Kinesis Data Firehose l'autorizzazione per inserire dati nel bucket Amazon S3.

   Per ulteriori informazioni, consulta Controllo degli accessi con Kinesis Data Firehose nella Guida per sviluppatori di Amazon Kinesis Data Firehose.

   In primo luogo, utilizza un editor di testo per creare una policy di attendibilità in un file `~/TrustPolicyForFirehose.json`, come segue:

   ```
   {
     "Statement": {
       "Effect": "Allow",
       "Principal": { "Service": "firehose.amazonaws.com" },
   ```

```
    "Action": "sts:AssumeRole"
    }
}
```

3. Utilizza il comando create-role per creare un ruolo IAM, specificando il file della policy di attendibilità. Annota il valore Role.Arn restituito, poiché ne avrai bisogno anche in una fase successiva:

```
aws iam create-role \
 --role-name FirehosetoS3Role \
 --assume-role-policy-document file://~/TrustPolicyForFirehose.json

{
    "Role": {
        "AssumeRolePolicyDocument": {
            "Statement": {
                "Action": "sts:AssumeRole",
                "Effect": "Allow",
                "Principal": {
                    "Service": "firehose.amazonaws.com"
                }
            }
        },
        "RoleId": "AAOIIAH450GAB4HC5F431",
        "CreateDate": "2015-05-29T13:46:29.431Z",
        "RoleName": "FirehosetoS3Role",
        "Path": "/",
        "Arn": "arn:aws:iam::123456789012:role/FirehosetoS3Role"
    }
}
```

4. Crea una policy di autorizzazioni per definire le operazioni che possono essere eseguite da Kinesis Data Firehose nel tuo account. In primo luogo, utilizza un editor di testo per creare una policy di autorizzazione in un file `~/PermissionsForFirehose.json`:

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
          "s3:AbortMultipartUpload",
          "s3:GetBucketLocation",
          "s3:GetObject",
```

```
            "s3:ListBucket",
            "s3:ListBucketMultipartUploads",
            "s3:PutObject" ],
        "Resource": [
            "arn:aws:s3:::my-bucket",
            "arn:aws:s3:::my-bucket/*" ]
      }
    ]
}
```

5. Associate la politica delle autorizzazioni al ruolo utilizzando il seguente comando: put-role-policy

```
aws iam put-role-policy --role-name FirehosetoS3Role --policy-name Permissions-
Policy-For-Firehose --policy-document file://~/PermissionsForFirehose.json
```

6. Crea un flusso di consegna Kinesis Data Firehose come mostrato di seguito, sostituendo i valori segnaposto per RoleARN e BucketARN con gli ARN del ruolo e del bucket creati:

```
aws firehose create-delivery-stream \
   --delivery-stream-name 'my-delivery-stream' \
   --s3-destination-configuration \
  '{"RoleARN": "arn:aws:iam::123456789012:role/FirehosetoS3Role", "BucketARN":
 "arn:aws:s3:::my-bucket"}'
```

Nota che Kinesis Data Firehose utilizza automaticamente un prefisso nel formato AAAA/MM/DD/HH UTC per gli oggetti Amazon S3 consegnati. Puoi specificare un ulteriore prefisso da aggiungere davanti al prefisso del formato data e ora. Se il prefisso termina con una barra (/), viene visualizzato come cartella nel bucket Amazon S3.

7. Attendi finché il flusso non si attiva (questa operazione potrebbe richiedere alcuni minuti). È possibile utilizzare il comando Kinesis describe-delivery-streamData Firehose per controllare il. DeliveryStreamDescription DeliveryStreamStatusproprietà. Inoltre, nota il DeliveryStreamDescription. DeliveryStreamValore ARN, in quanto sarà necessario in un passaggio successivo:

```
aws firehose describe-delivery-stream --delivery-stream-name "my-delivery-stream"
{
    "DeliveryStreamDescription": {
        "HasMoreDestinations": false,
        "VersionId": "1",
        "CreateTimestamp": 1446075815.822,
```

```
        "DeliveryStreamARN": "arn:aws:firehose:us-
east-1:123456789012:deliverystream/my-delivery-stream",
        "DeliveryStreamStatus": "ACTIVE",
        "DeliveryStreamName": "my-delivery-stream",
        "Destinations": [
            {
                "DestinationId": "destinationId-000000000001",
                "S3DestinationDescription": {
                    "CompressionFormat": "UNCOMPRESSED",
                    "EncryptionConfiguration": {
                        "NoEncryptionConfig": "NoEncryption"
                    },
                    "RoleARN": "delivery-stream-role",
                    "BucketARN": "arn:aws:s3:::my-bucket",
                    "BufferingHints": {
                        "IntervalInSeconds": 300,
                        "SizeInMBs": 5
                    }
                }
            }
        ]
    }
}
```

8. Crea il ruolo IAM che concede a CloudWatch Logs l'autorizzazione a inserire i dati nel flusso di distribuzione di Kinesis Data Firehose. In primo luogo, utilizza un editor di testo per creare una policy di attendibilità in un file `~/TrustPolicyForCWL.json`:

   Questa policy include una chiave di contesto della condizione globale `aws:SourceArn` per prevenire il problema di sicurezza noto come "confused deputy". Per ulteriori informazioni, consulta Prevenzione del "confused deputy".

```
{
  "Statement": {
    "Effect": "Allow",
    "Principal": { "Service": "logs.amazonaws.com" },
    "Action": "sts:AssumeRole",
    "Condition": {
         "StringLike": {
             "aws:SourceArn": "arn:aws:logs:region:123456789012:*"
         }
     }
  }
```

```
}
```

9. Utilizza il comando create-role per creare un ruolo IAM, specificando il file della policy di attendibilità. Annota il valore Role.Arn restituito, poiché ne avrai bisogno anche in una fase successiva:

```
aws iam create-role \
--role-name CWLtoKinesisFirehoseRole \
--assume-role-policy-document file://~/TrustPolicyForCWL.json

{
    "Role": {
        "AssumeRolePolicyDocument": {
            "Statement": {
                "Action": "sts:AssumeRole",
                "Effect": "Allow",
                "Principal": {
                    "Service": "logs.amazonaws.com"
                },
                "Condition": {
                     "StringLike": {
                         "aws:SourceArn": "arn:aws:logs:region:123456789012:*"
                     }
                 }
            }
        },
        "RoleId": "AAOIIAH450GAB4HC5F431",
        "CreateDate": "2015-05-29T13:46:29.431Z",
        "RoleName": "CWLtoKinesisFirehoseRole",
        "Path": "/",
        "Arn": "arn:aws:iam::123456789012:role/CWLtoKinesisFirehoseRole"
    }
}
```

10. Crea una politica di autorizzazioni per definire quali azioni i CloudWatch log possono eseguire sul tuo account. In primo luogo, utilizza un editor di testo per creare un file di policy di autorizzazioni (ad esempio, ~/PermissionsForCWL.json):

```
{
    "Statement":[
      {
        "Effect":"Allow",
        "Action":["firehose:PutRecord"],
```

```
        "Resource":[
            "arn:aws:firehose:region:account-id:deliverystream/delivery-stream-
name"]
      }
    ]
}
```

11. Associa la politica delle autorizzazioni al ruolo utilizzando il comando: put-role-policy

```
aws iam put-role-policy --role-name CWLtoKinesisFirehoseRole --policy-
name Permissions-Policy-For-CWL --policy-document file://~/PermissionsForCWL.json
```

12. Dopo che il flusso di distribuzione di Amazon Kinesis Data Firehose è attivo e hai creato il ruolo IAM, puoi CloudWatch creare il filtro di sottoscrizione Logs. Il filtro di sottoscrizione avvia immediatamente il flusso di dati di log in tempo reale dal gruppo di log selezionato nel flusso di consegna Amazon Kinesis Data Firehose:

```
aws logs put-subscription-filter \
    --log-group-name "CloudTrail" \
    --filter-name "Destination" \
    --filter-pattern "{$.userIdentity.type = Root}" \
    --destination-arn "arn:aws:firehose:region:123456789012:deliverystream/my-
delivery-stream" \
    --role-arn "arn:aws:iam::123456789012:role/CWLtoKinesisFirehoseRole"
```

13. Dopo aver configurato il filtro di abbonamento, CloudWatch Logs inoltrerà tutti gli eventi di log in entrata che corrispondono al modello di filtro al flusso di distribuzione di Amazon Kinesis Data Firehose. I dati inizieranno a essere visualizzati in Amazon S3 in base all'intervallo del buffer di tempo impostato nel flusso di consegna Amazon Kinesis Data Firehose. Quando è trascorso tempo sufficiente, puoi verificare i dati controllando il Bucket Amazon S3.

```
aws s3api list-objects --bucket 'my-bucket' --prefix 'firehose/'
{
    "Contents": [
        {
            "LastModified": "2015-10-29T00:01:25.000Z",
            "ETag": "\"a14589f8897f4089d3264d9e2d1f1610\"",
            "StorageClass": "STANDARD",
            "Key": "firehose/2015/10/29/00/my-delivery-stream-2015-10-29-00-01-21-
a188030a-62d2-49e6-b7c2-b11f1a7ba250",
            "Owner": {
                "DisplayName": "cloudwatch-logs",
```

```
                "ID": "1ec9cf700ef6be062b19584e0b7d84ecc19237f87b5"
            },
            "Size": 593
        },
        {
            "LastModified": "2015-10-29T00:35:41.000Z",
            "ETag": "\"a7035b65872bb2161388ffb63dd1aec5\"",
            "StorageClass": "STANDARD",
            "Key": "firehose/2015/10/29/00/my-delivery-
stream-2015-10-29-00-35-40-7cc92023-7e66-49bc-9fd4-fc9819cc8ed3",
            "Owner": {
                "DisplayName": "cloudwatch-logs",
                "ID": "1ec9cf700ef6be062b19584e0b7d84ecc19237f87b6"
            },
            "Size": 5752
        }
    ]
}
```

```
aws s3api get-object --bucket 'my-bucket' --key 'firehose/2015/10/29/00/my-
delivery-stream-2015-10-29-00-01-21-a188030a-62d2-49e6-b7c2-b11f1a7ba250'
 testfile.gz

{
    "AcceptRanges": "bytes",
    "ContentType": "application/octet-stream",
    "LastModified": "Thu, 29 Oct 2015 00:07:06 GMT",
    "ContentLength": 593,
    "Metadata": {}
}
```

I dati nell'oggetto di Amazon S3 vengono compressi nel formato gzip. Puoi esaminare i dati non elaborati dalla riga di comando utilizzando i seguenti comandi Unix:

```
zcat testfile.gz
```

Condivisione di dati di log tra più account con le sottoscrizioni

Puoi collaborare con il proprietario di un altro AWS account e ricevere i relativi eventi di registro sulle tue AWS risorse, ad esempio uno stream Amazon Kinesis o Amazon Kinesis Data Firehose

(questa operazione è nota come condivisione di dati tra account). Ad esempio, questi dati di log eventi possono essere letti da un flusso Kinesis Data Streams o Kinesis Data Firehose centralizzato per eseguire elaborazione e analisi personalizzate. L'elaborazione personalizzata è particolarmente utile quando collabori e analizzi dati tra più account.

Ad esempio, il gruppo di sicurezza di informazioni di un'azienda, potrebbe voler analizzare i dati per il rilevamento delle intrusioni in tempo reale o i comportamenti anomali, per poter condurre un'ispezione degli account in tutte le divisioni dell'azienda raccogliendo i log di produzione federata per l'elaborazione centralizzata. Un flusso di dati di evento in tempo reale tra questi account può essere assemblato e distribuito ai gruppi di sicurezza di informazioni che possono utilizzare Kinesis Data Streams per collegare i dati ai propri sistemi di analisi di sicurezza esistenti.

Argomenti

Condivisione di dati di log tra più account tramite Kinesis Data Streams

Durante la creazione di una sottoscrizione tra più account, è possibile specificare un singolo account o un'organizzazione come mittente. Nel caso in cui si specifichi un'organizzazione, la procedura illustrata di seguito consente a tutti gli account dell'organizzazione di inviare log all'account del destinatario.

Per condividere dati di log tra diversi account, è necessario stabilire un mittente e un ricevitore di dati di log:

- Registra il mittente dei dati: ottiene le informazioni sulla destinazione dal destinatario e comunica a CloudWatch Logs che è pronto a inviare gli eventi di registro alla destinazione specificata. Nelle procedure illustrate nel resto di questa sezione, il mittente dei dati di registro viene visualizzato con un numero di account fittizio pari a 1111 AWS .

  Nel caso in cui più account all'interno di un'organizzazione inviano log a un account del destinatario, è possibile creare una policy che conceda a tutti gli account dell'organizzazione l'autorizzazione per eseguire tale operazione. Devi comunque impostare filtri di sottoscrizione separati per ciascun account del mittente.
- Destinatario dei dati di registro: imposta una destinazione che incapsula un flusso Kinesis Data Streams e CloudWatch fa sapere a Logs che il destinatario desidera ricevere i dati di registro. Il

destinatario quindi condivide le informazioni su questa destinazione con il mittente. Nelle procedure illustrate nel resto di questa sezione, il destinatario dei dati di registro viene mostrato con un numero di account fittizio pari a 9999. AWS

Per iniziare a ricevere gli eventi di registro da utenti con più account, il destinatario dei dati di registro crea innanzitutto una destinazione Logs. CloudWatch Ogni destinazione è formata dai seguenti elementi chiave:

Nome della destinazione

Il nome della destinazione che intendi creare.

ARN di destinazione

L'Amazon Resource Name (ARN) della AWS risorsa che desideri utilizzare come destinazione del feed di abbonamento.

ARN del ruolo

Un ruolo AWS Identity and Access Management (IAM) che concede a CloudWatch Logs le autorizzazioni necessarie per inserire i dati nel flusso scelto.

Policy di accesso

Un documento della policy IAM (in formato JSON, scritto utilizzando la grammatica delle policy IAM) che controlla l'insieme degli utenti ai quali è concesso scrivere nella tua destinazione.

Il gruppo di log e la destinazione devono trovarsi nella stessa regione. AWS Tuttavia, la risorsa AWS a cui punta la destinazione può trovarsi in una Regione diversa. Negli esempi delle sezioni seguenti, tutte le risorse specifiche della Regione vengono create in Stati Uniti orientali (Virginia settentrionale).

Argomenti

Configurazione di una nuova sottoscrizione tra più account

Segui la procedura riportata in queste sezioni per configurare una nuova sottoscrizione del log tra più account.

Argomenti

Passaggio 1: creazione di una destinazione

> **Important**
>
> Tutte le fasi di questa procedura devono essere eseguite nell'account del destinatario dei dati di log.

Per questo esempio, l'account del destinatario dei dati di registro ha un ID AWS account di 9999, mentre l'ID dell'account mittente AWS dei dati di registro è 1111.

Questo esempio crea una destinazione utilizzando un flusso Kinesis Data RecipientStream Streams chiamato e un ruolo CloudWatch che consente a Logs di scrivere dati su di esso.

Quando viene creata la destinazione, CloudWatch Logs invia un messaggio di prova alla destinazione per conto dell'account del destinatario. Quando il filtro di sottoscrizione è attivo in un secondo momento, CloudWatch Logs invia gli eventi di registro alla destinazione per conto dell'account di origine.

Creazione di una destinazione

1. Nell'account del destinatario, crea un flusso di destinazione in Kinesis Data Streams. Al prompt dei comandi, digita:

   ```
   aws kinesis create-stream --stream-name "RecipientStream" --shard-count 1
   ```

2. Attendi finché il flusso non diventa attivo. Puoi usare il comando aws kinesis describe-stream per controllare il. StreamDescription StreamStatusproprietà. Inoltre, prendi nota del valore StreamDescription.StreamArn perché lo passerai a Logs in un secondo CloudWatch momento:

```
aws kinesis describe-stream --stream-name "RecipientStream"
{
  "StreamDescription": {
    "StreamStatus": "ACTIVE",
    "StreamName": "RecipientStream",
    "StreamARN": "arn:aws:kinesis:us-east-1:999999999999:stream/RecipientStream",
    "Shards": [
      {
        "ShardId": "shardId-000000000000",
        "HashKeyRange": {
          "EndingHashKey": "340282366920938463463374607431768EXAMPLE",
          "StartingHashKey": "0"
        },
        "SequenceNumberRange": {
          "StartingSequenceNumber":
 "49551135218688818456679503831981458784591352702 18EXAMPLE"
        }
      }
    ]
  }
}
```

Potrebbero essere necessari uno o due minuti perché il flusso sia in stato attivo.

3. Crea il ruolo IAM che concede a CloudWatch Logs l'autorizzazione a inserire dati nel tuo stream. Per prima cosa, devi creare una politica di fiducia in un file TrustPolicyFor~/ CWL.json. Utilizza un editor di testo per creare questo file di policy, non utilizzare la console IAM.

   Questa policy include una chiave di contesto della condizione globale `aws:SourceArn` che specifica il `sourceAccountId` per prevenire il problema di sicurezza noto come "confused deputy". Se non conosci ancora l'ID dell'account di origine nella prima chiamata, consigliamo di inserire l'ARN di destinazione nel campo ARN di origine. Nelle chiamate successive, è necessario impostare l'ARN di origine come l'ARN di origine effettivo raccolto dalla prima chiamata. Per ulteriori informazioni, consulta Prevenzione del "confused deputy".

```
{
    "Statement": {
        "Effect": "Allow",
        "Principal": {
            "Service": "logs.amazonaws.com"
        },
```

```
        "Condition": {
            "StringLike": {
                "aws:SourceArn": [
                    "arn:aws:logs:region:sourceAccountId:*",
                    "arn:aws:logs:region:recipientAccountId:*"
                ]
            }
        },
        "Action": "sts:AssumeRole"
    }
}
```

4. Utilizza il comando aws iam create-role per creare il ruolo IAM, specificando il file della policy di attendibilità. Prendi nota del valore Role.Arn restituito perché verrà passato anche a Logs in un secondo momento: CloudWatch

```
aws iam create-role \
--role-name CWLtoKinesisRole \
--assume-role-policy-document file://~/TrustPolicyForCWL.json

{
    "Role": {
        "AssumeRolePolicyDocument": {
            "Statement": {
                "Action": "sts:AssumeRole",
                "Effect": "Allow",
                "Condition": {
                    "StringLike": {
                        "aws:SourceArn": [
                            "arn:aws:logs:region:sourceAccountId:*",
                            "arn:aws:logs:region:recipientAccountId:*"
                        ]
                    }
                },
                "Principal": {
                    "Service": "logs.amazonaws.com"
                }
            }
        },
        "RoleId": "AAOIIAH450GAB4HC5F431",
        "CreateDate": "2015-05-29T13:46:29.431Z",
        "RoleName": "CWLtoKinesisRole",
        "Path": "/",
```

```
        "Arn": "arn:aws:iam::999999999999:role/CWLtoKinesisRole"
    }
}
```

5. Crea una politica di autorizzazioni per definire quali azioni i CloudWatch log possono eseguire sul tuo account. Innanzitutto, usa un editor di testo per creare una politica di autorizzazioni in un file ~/ CWL.json: PermissionsFor

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "kinesis:PutRecord",
      "Resource": "arn:aws:kinesis:region:999999999999:stream/RecipientStream"
    }
  ]
}
```

6. Associa la politica delle autorizzazioni al ruolo utilizzando il comando aws iam: put-role-policy

```
aws iam put-role-policy \
    --role-name CWLtoKinesisRole \
    --policy-name Permissions-Policy-For-CWL \
    --policy-document file://~/PermissionsForCWL.json
```

7. Dopo che lo stream è nello stato attivo e hai creato il ruolo IAM, puoi creare la destinazione CloudWatch Logs.

   a. In questa fase alla tua destinazione non si associa una policy d'accesso predefinita e costituisce solo la prima delle due fasi necessarie per completare la creazione della destinazione. Prendi nota di DestinationArnciò che viene restituito nel payload:

```
aws logs put-destination \
    --destination-name "testDestination" \
    --target-arn "arn:aws:kinesis:region:999999999999:stream/RecipientStream" \
    --role-arn "arn:aws:iam::999999999999:role/CWLtoKinesisRole"

{
  "DestinationName" : "testDestination",
  "RoleArn" : "arn:aws:iam::999999999999:role/CWLtoKinesisRole",
  "DestinationArn" : "arn:aws:logs:us-
east-1:999999999999:destination:testDestination",
```

```
  "TargetArn" : "arn:aws:kinesis:us-east-1:999999999999:stream/RecipientStream"
}
```

b. Dopo aver completato la fase 7, nell'account del destinatario dei dati di log associa alla destinazione una policy d'accesso predefinita. Questa politica deve specificare i log: PutSubscriptionFilter action e concede l'autorizzazione all'account mittente di accedere alla destinazione.

La politica concede l'autorizzazione all' AWS account che invia i log. Puoi specificare solo questo account nella policy oppure, se l'account del mittente è membro di un'organizzazione, la policy può specificare l'ID dell'organizzazione. In questo modo, puoi creare una sola policy per consentire a più account di un'organizzazione di inviare log a questo account di destinazione.

Utilizza un editor di testo per creare un file denominato `~/AccessPolicy.json` con una delle seguenti istruzioni di policy.

Questa prima policy di esempio consente a tutti gli account dell'organizzazione che hanno un ID di `o-1234567890` di inviare log all'account del destinatario.

```
{
    "Version" : "2012-10-17",
    "Statement" : [
        {
            "Sid" : "",
            "Effect" : "Allow",
            "Principal" : "*",
            "Action" : "logs:PutSubscriptionFilter",
            "Resource" :
 "arn:aws:logs:region:999999999999:destination:testDestination",
            "Condition": {
               "StringEquals" : {
                   "aws:PrincipalOrgID" : ["o-1234567890"]
                }
            }
        }
    ]
}
```

Nell'esempio seguente, solo l'account del mittente dei dati di log (111111111111) può inviare log all'account del destinatario dei dati di log.

```
{
  "Version" : "2012-10-17",
  "Statement" : [
    {
      "Sid" : "",
      "Effect" : "Allow",
      "Principal" : {
        "AWS" : "111111111111"
      },
      "Action" : "logs:PutSubscriptionFilter",
      "Resource" :
 "arn:aws:logs:region:999999999999:destination:testDestination"
    }
  ]
}
```

c. Collega la policy creata nel passaggio precedente alla destinazione.

```
aws logs put-destination-policy \
    --destination-name "testDestination" \
    --access-policy file://~/AccessPolicy.json
```

Questa politica di accesso consente agli utenti dell' AWS account con ID 1111 di effettuare chiamate ***PutSubscriptionFilter*** *verso la destinazione con ARN arn:aws:logs:region:99999999:destination:testDestination.* Qualsiasi tentativo di chiamata di un altro utente verso questa destinazione verrà rifiutato. PutSubscriptionFilter

Per convalidare i privilegi di un utente su una policy d'accesso predefinita, consulta Utilizzo dello strumento di validazione delle policy nella guida per l'utente IAM.

Al termine, se utilizzi le autorizzazioni AWS Organizations per più account, segui la procedura riportata di seguito. Fase 2: creazione di un ruolo IAM (solo se si utilizza un'organizzazione) Se le autorizzazioni vengono concesse direttamente all'altro account anziché utilizzare Organizations, puoi saltare tale passaggio e procedere alla sezione Passaggio 4: creazione di un filtro di sottoscrizione.

Fase 2: creazione di un ruolo IAM (solo se si utilizza un'organizzazione)

Se nella sezione precedente hai creato la destinazione utilizzando una policy di accesso che concede le autorizzazioni all'organizzazione in cui è presente l'account 111111111111, invece di concederle

direttamente all'account 111111111111, segui i passaggi descritti di seguito. In caso contrario, passa alla sezione Passaggio 4: creazione di un filtro di sottoscrizione.

I passaggi descritti in questa sezione creano un ruolo IAM, che CloudWatch può presupporre e verificare se l'account mittente è autorizzato a creare un filtro di sottoscrizione in base alla destinazione del destinatario.

Per l'account mittente, segui la procedura descritta in questa sezione. Il ruolo deve esistere nell'account mittente e devi specificare l'ARN di questo ruolo nel filtro di sottoscrizione. In questo esempio, l'account mittente è denominato 111111111111.

Creazione del ruolo IAM necessario per le sottoscrizioni del log tra più account utilizzando AWS Organizations

1. Crea la policy di attendibilità seguente in un file `/TrustPolicyForCWLSubscriptionFilter.json`. Utilizza un editor di testo per creare questo file di policy; non utilizzare la console IAM.

```
{
  "Statement": {
    "Effect": "Allow",
    "Principal": { "Service": "logs.amazonaws.com" },
    "Action": "sts:AssumeRole"
  }
}
```

2. Crea il ruolo IAM che utilizza questa policy. Prendi nota del valore `Arn` restituito dal comando, sarà necessario in seguito in questa procedura. In questo esempio, il ruolo in fase di creazione è denominato `CWLtoSubscriptionFilterRole`.

```
aws iam create-role \
     --role-name CWLtoSubscriptionFilterRole \
     --assume-role-policy-document file://~/
TrustPolicyForCWLSubscriptionFilter.json
```

3. Crea una politica di autorizzazioni per definire le azioni che CloudWatch Logs può eseguire sul tuo account.

   a. In primo luogo, utilizza un editor di testo per creare la policy di autorizzazione seguente in un file denominato `~/PermissionsForCWLSubscriptionFilter.json`.

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "logs:PutLogEvents",
            "Resource": "arn:aws:logs:region:111111111111:log-
group:LogGroupOnWhichSubscriptionFilterIsCreated:*"
        }
    ]
}
```

b. Inserisci il comando seguente per associare la policy di autorizzazione appena creata al ruolo creato nella fase 2.

```
aws iam put-role-policy
    --role-name CWLtoSubscriptionFilterRole
    --policy-name Permissions-Policy-For-CWL-Subscription-filter
    --policy-document file://~/PermissionsForCWLSubscriptionFilter.json
```

Al termine dell'operazione, passa alla sezione Passaggio 4: creazione di un filtro di sottoscrizione.

Passaggio 3: aggiunta/convalida delle autorizzazioni IAM per la destinazione multi-account

In base alla AWS logica di valutazione dei criteri tra account, per accedere a qualsiasi risorsa tra account (ad esempio uno stream Kinesis o Kinesis Data Firehose utilizzato come destinazione per un filtro di sottoscrizione) è necessario disporre di una politica basata sull'identità nell'account di invio che fornisca l'accesso esplicito alla risorsa di destinazione tra account diversi. Per ulteriori informazioni sulla logica di valutazione delle policy, consulta la pagina Cross-account policy evaluation logic.

Puoi collegare la policy basata sull'identità al ruolo IAM o all'utente IAM che stai utilizzando per creare il filtro di sottoscrizione. Questa policy deve essere presente nell'account mittente. Se utilizzi il ruolo di amministratore per creare il filtro di sottoscrizione, puoi saltare questo passaggio e passare a Passaggio 4: creazione di un filtro di sottoscrizione.

Aggiunta o convalida delle autorizzazioni IAM necessarie per più account

1. Inserisci il seguente comando per verificare quale ruolo IAM o utente IAM viene utilizzato per eseguire i comandi di log AWS .

```
aws sts get-caller-identity
```

Il comando restituisce un output simile al seguente:

```
{
"UserId": "User ID",
"Account": "sending account id",
"Arn": "arn:aws:sending account id:role/user:RoleName/UserName"
}
```

Prendi nota del valore rappresentato da o. *RoleNameUserName*

2. AWS Management Console Accedi all'account di invio e cerca le policy allegate con il ruolo IAM o l'utente IAM restituito nell'output del comando che hai inserito nel passaggio 1.
3. Verifica che le policy associate a questo ruolo o utente forniscano autorizzazioni esplicite per richiamare `logs:putSubscriptionFilter` sulla risorsa di destinazione multi-account. Le seguenti policy di esempio mostrano le autorizzazioni suggerite.

   La seguente politica fornisce le autorizzazioni per creare un filtro di sottoscrizione su qualsiasi risorsa di destinazione solo in un singolo AWS account, account123456789012:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Allow subscription filters on any resource in one specific
 account",
            "Effect": "Allow",
            "Action": "logs:PutSubscriptionFilter",
            "Resource": [
                "arn:aws:logs:*:*:log-group:*",
                "arn:aws:logs:*:123456789012:destination:*"
            ]
        }
    ]
}
```

La seguente politica fornisce le autorizzazioni per creare un filtro di sottoscrizione solo su una risorsa di destinazione specifica denominata `sampleDestination` in AWS account singolo, account: `123456789012`

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Allow subscription filters on one specific resource in one
 specific account",
            "Effect": "Allow",
            "Action": "logs:PutSubscriptionFilter",
            "Resource": [
                "arn:aws:logs:*:*:log-group:*",
                "arn:aws:logs:*:123456789012:destination:sampleDestination"
            ]
        }
    ]
}
```

Passaggio 4: creazione di un filtro di sottoscrizione

Una volta creata una destinazione, l'account del destinatario dei dati di log può condividere l'ARN di destinazione (arn:aws:logs:us-east-1:999999999999:destination:testDestination) con altri account AWS , perché questi possano inviare eventi di log alla stessa destinazione. Tali utenti di questi account di invio possono creare un filtro di sottoscrizione sui rispettivi gruppi di log sulla destinazione. Il filtro di sottoscrizione avvia immediatamente il flusso di dati di log in tempo reale dal gruppo di log selezionato alla destinazione specificata.

> **Note**
>
> Se stai concedendo le autorizzazioni per il filtro di sottoscrizione a un'intera organizzazione, dovrai utilizzare l'ARN del ruolo IAM che hai creato in Fase 2: creazione di un ruolo IAM (solo se si utilizza un'organizzazione).

Nell'esempio seguente, viene creato un filtro di sottoscrizione in un account di invio. Il filtro è associato a un gruppo di log contenente AWS CloudTrail eventi in modo che ogni attività registrata

effettuata dalle AWS credenziali «Root» venga consegnata alla destinazione creata in precedenza. Tale destinazione incapsula un flusso chiamato "». RecipientStream

Il resto dei passaggi descritti nelle sezioni seguenti presuppone che l'utente abbia seguito le istruzioni riportate in Invio di CloudTrail eventi ai CloudWatch registri nella Guida per l'AWS CloudTrail utente e abbia creato un gruppo di log contenente gli eventi. CloudTrail Questi passaggi presuppongono che il nome di questo gruppo di log sia `CloudTrail/logs`.

Quando inserisci il comando seguente, assicurati di aver effettuato l'accesso come utente IAM o di utilizzare il ruolo IAM per cui hai aggiunto la policy in Passaggio 3: aggiunta/convalida delle autorizzazioni IAM per la destinazione multi-account.

```
aws logs put-subscription-filter \
    --log-group-name "CloudTrail/logs" \
    --filter-name "RecipientStream" \
    --filter-pattern "{$.userIdentity.type = Root}" \
    --destination-arn "arn:aws:logs:region:999999999999:destination:testDestination"
```

Il gruppo di log e la destinazione devono trovarsi nella stessa AWS regione. Tuttavia, la destinazione può puntare a una AWS risorsa come un flusso Kinesis Data Streams che si trova in una regione diversa.

Convalida del flusso dei log eventi

Dopo aver creato il filtro di abbonamento, CloudWatch Logs inoltra tutti gli eventi di registro in entrata che corrispondono al modello di filtro allo stream incapsulato nel flusso di destinazione denominato "». RecipientStream Il proprietario della destinazione può verificare che ciò stia accadendo utilizzando il get-shard-iterator comando aws kinesis per acquisire uno shard Kinesis Data Streams e utilizzando il comando aws kinesis get-records per recuperare alcuni record Kinesis Data Streams:

```
aws kinesis get-shard-iterator \
      --stream-name RecipientStream \
      --shard-id shardId-000000000000 \
      --shard-iterator-type TRIM_HORIZON

{
    "ShardIterator":
    "AAAAAAAAAAFGU/
kLvNggvndHq2UIFOw5PZc6F01s3e3afsSscRM70JSbjIefg2ub07nk1y6CDxYR1UoGHJNP4m4NFUetzfL+wev
+e2P4djJg4L9wmXKvQYoE+rMUiFq+p4Cn3IgvqOb5dRA0yybNdRcdzvnC35KQANoHzzahKdRGb9v4scv+3vaq+f
+OIK8zM5My8ID+g6rMo7UKWeI4+IWiKEXAMPLE"
```

```
}

aws kinesis get-records \
      --limit 10 \
      --shard-iterator
      "AAAAAAAAAAFGU/
kLvNggvndHq2UIFOw5PZc6F01s3e3afsSscRM70JSbjIefg2ub07nk1y6CDxYR1UoGHJNP4m4NFUetzfL+wev
+e2P4djJg4L9wmXKvQYoE+rMUiFq+p4Cn3IgvqOb5dRA0yybNdRcdzvnC35KQANoHzzahKdRGb9v4scv+3vaq+f
+OIK8zM5My8ID+g6rMo7UKWeI4+IWiKEXAMPLE"
```

> **Note**
>
> Potresti dover rieseguire il comando get-records più volte prima che il flusso Kinesis Data Streams inizi a restituire dati.

Dovresti visualizzare una risposta con un array di log di Kinesis Data Streams. L'attributo dati nel log di Kinesis Data Streams usa la compressione nel formato gzip e la codifica base64. Puoi esaminare i dati non elaborati dalla riga di comando utilizzando i seguenti comandi Unix:

```
echo -n "<Content of Data>" | base64 -d | zcat
```

I dati con codifica base64 e decompressi sono in formato JSON con la seguente struttura:

```
{
    "owner": "111111111111",
    "logGroup": "CloudTrail/logs",
    "logStream": "111111111111_CloudTrail/logs_us-east-1",
    "subscriptionFilters": [
        "RecipientStream"
    ],
    "messageType": "DATA_MESSAGE",
    "logEvents": [
        {
            "id": "31953106606966983378809025079804211143289615424 29EXAMPLE",
            "timestamp": 1432826855000,
            "message": "{\"eventVersion\":\"1.03\",\"userIdentity\":{\"type\":\"Root
\"}"
        },
        {
            "id": "31953106606966983378809025079804211143289615424 29EXAMPLE",
```

```
            "timestamp": 1432826855000,
            "message": "{\"eventVersion\":\"1.03\",\"userIdentity\":{\"type\":\"Root
\"}"
        },
        {
            "id": "31953106606966983378809025079804211143289615424 29EXAMPLE",
            "timestamp": 1432826855000,
            "message": "{\"eventVersion\":\"1.03\",\"userIdentity\":{\"type\":\"Root
\"}"
        }
    ]
}
```

Gli elementi chiave della struttura di dati sono i seguenti:

owner

AWS L'ID dell'account dei dati di registro di origine.

logGroup

Nome del gruppo di log dei dati di log originari.

logStream

Nome del flusso di log dei dati di log originari.

subscriptionFilters

Elenco dei nomi di filtro sottoscrizione che corrispondono con i dati di log originari.

messageType

I messaggi di dati usano il tipo "DATA_MESSAGE". A volte CloudWatch i log possono emettere record Kinesis Data Streams di tipo «CONTROL_MESSAGE», principalmente per verificare se la destinazione è raggiungibile.

logEvents

I dati di log effettivi, rappresentati come una varietà di record di eventi di log. La proprietà ID è un identificatore univoco per ogni eventi di log.

Modifica dell'appartenenza alla destinazione in fase di runtime

Potrebbero verificarsi situazioni in cui potresti dover aggiungere o rimuovere l'adesione di alcuni utenti da una destinazione da te posseduta. Puoi utilizzare il commando `put-destination-`

`policy` sulla destinazione con una nuova policy di accesso. Nell'esempio seguente, si impedisce a un account aggiunto in precedenza 111111111111 di inviare ulteriori dati di log, mentre l'account 222222222222 viene abilitato.

1. Recupera la politica attualmente associata alla destinazione TestDestination e prendi nota di: AccessPolicy

```
aws logs describe-destinations \
    --destination-name-prefix "testDestination"

{
 "Destinations": [
   {
     "DestinationName": "testDestination",
     "RoleArn": "arn:aws:iam::999999999999:role/CWLtoKinesisRole",
     "DestinationArn":
 "arn:aws:logs:region:999999999999:destination:testDestination",
     "TargetArn": "arn:aws:kinesis:region:999999999999:stream/RecipientStream",
     "AccessPolicy": "{\"Version\": \"2012-10-17\", \"Statement\":
 [{\"Sid\": \"\", \"Effect\": \"Allow\", \"Principal\": {\"AWS\":
 \"111111111111\"}, \"Action\": \"logs:PutSubscriptionFilter\", \"Resource\":
 \"arn:aws:logs:region:999999999999:destination:testDestination\"}] }"
   }
 ]
}
```

2. Aggiorna la policy per riflettere l'arresto di tale account 111111111111, mentre l'account 222222222222 viene abilitato. Inserisci questa politica nel file ~/ .json: NewAccessPolicy

```
{
  "Version" : "2012-10-17",
  "Statement" : [
    {
      "Sid" : "",
      "Effect" : "Allow",
      "Principal" : {
        "AWS" : "222222222222"
      },
      "Action" : "logs:PutSubscriptionFilter",
      "Resource" : "arn:aws:logs:region:999999999999:destination:testDestination"
    }
  ]
```

```
}
```

3. Chiama PutDestinationPolicyper associare la politica definita nel NewAccessPolicyfile.json alla destinazione:

```
aws logs put-destination-policy \
--destination-name "testDestination" \
--access-policy file://~/NewAccessPolicy.json
```

Alla fine ciò disabiliterà gli eventi di log dall'ID account 111111111111. I log eventi provenienti dall'ID account 222222222222 iniziano a fluire verso la destinazione non appena il proprietario dell'account 222222222222 crea un filtro di sottoscrizione.

Aggiornamento di una sottoscrizione tra più account esistente

Se hai una sottoscrizione del log tra più account in cui l'account di destinazione concede le autorizzazioni solo a specifici account del mittente e desideri aggiornare questa sottoscrizione in modo che l'account di destinazione conceda l'accesso a tutti gli account di un'organizzazione, segui la procedura descritta in questa sezione.

Argomenti

Fase 1: aggiornamento dei filtri di sottoscrizione

> **Note**
>
> Questo passaggio è necessario solo per le sottoscrizioni tra più account per i log creati dai servizi elencati in Abilitazione della registrazione dai servizi AWS. Se non stai lavorando con log creati da uno di questi gruppi di log, puoi passare alla sezione Fase 2: aggiornamento della policy di accesso alla destinazione esistente.

In alcuni casi, devi aggiornare i filtri di sottoscrizione in tutti gli account del mittente che inviano log all'account di destinazione. L'aggiornamento aggiunge un ruolo IAM, che CloudWatch può presupporre e convalidare che l'account mittente sia autorizzato a inviare i log all'account del destinatario.

Segui la procedura descritta in questa sezione per ogni account del mittente che desideri aggiornare in modo da utilizzare l'ID dell'organizzazione per le autorizzazioni di sottoscrizione tra più account.

Negli esempi di questa sezione sono già stati creati dei filtri di sottoscrizione negli account 111111111111 e 222222222222 per l'invio di log all'account 999999999999. I valori del filtro di sottoscrizione esistenti sono i seguenti:

```
## Existing Subscription Filter parameter values
    \ --log-group-name "my-log-group-name"
    \ --filter-name "RecipientStream"
    \ --filter-pattern "{$.userIdentity.type = Root}"
    \ --destination-arn "arn:aws:logs:region:999999999999:destination:testDestination"
```

Se è necessario trovare i valori dei parametri del filtro di sottoscrizione correnti, digita il comando seguente.

```
aws logs describe-subscription-filters
    \ --log-group-name "my-log-group-name"
```

Aggiornamento di un filtro di sottoscrizione per iniziare a utilizzare gli ID dell'organizzazione per le autorizzazioni del log tra più account

1. Crea la policy di attendibilità seguente in un file `~/TrustPolicyForCWL.json`. Utilizza un editor di testo per creare questo file di policy; non utilizzare la console IAM.

   ```
   {
     "Statement": {
       "Effect": "Allow",
       "Principal": { "Service": "logs.amazonaws.com" },
       "Action": "sts:AssumeRole"
     }
   }
   ```

2. Crea il ruolo IAM che utilizza questa policy. Prendi nota del valore `Arn` del valore `Arn` restituito dal comando, sarà necessario in seguito in questa procedura. In questo esempio, il ruolo in fase di creazione è denominato `CWLtoSubscriptionFilterRole`.

   ```
   aws iam create-role
       \ --role-name CWLtoSubscriptionFilterRole
       \ --assume-role-policy-document file://~/TrustPolicyForCWL.json
   ```

3. Crea una politica di autorizzazioni per definire le azioni che CloudWatch Logs può eseguire sul tuo account.

   a. In primo luogo, utilizza un editor di testo per creare la policy di autorizzazione seguente in un file denominato `/PermissionsForCWLSubscriptionFilter.json`.

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "logs:PutLogEvents",
            "Resource": "arn:aws:logs:region:111111111111:log-
group:LogGroupOnWhichSubscriptionFilterIsCreated:*"
        }
    ]
}
```

   b. Inserisci il comando seguente per associare la policy di autorizzazione appena creata al ruolo creato nella fase 2.

```
aws iam put-role-policy
    --role-name CWLtoSubscriptionFilterRole
    --policy-name Permissions-Policy-For-CWL-Subscription-filter
    --policy-document file://~/PermissionsForCWLSubscriptionFilter.json
```

4. Inserisci il comando seguente per aggiornare il filtro di sottoscrizione.

```
aws logs put-subscription-filter
    \ --log-group-name "my-log-group-name"
    \ --filter-name "RecipientStream"
    \ --filter-pattern "{$.userIdentity.type = Root}"
    \ --destination-arn
 "arn:aws:logs:region:999999999999:destination:testDestination"
    \ --role-arn "arn:aws:iam::111111111111:role/CWLtoSubscriptionFilterRole"
```

Fase 2: aggiornamento della policy di accesso alla destinazione esistente

Dopo aver aggiornato i filtri di sottoscrizione in tutti gli account del mittente, è possibile aggiornare la policy di accesso alla destinazione nell'account del destinatario.

Negli esempi seguenti, l'account del destinatario è `999999999999` e la destinazione è denominata `testDestination`.

L'aggiornamento abilita tutti gli account che fanno parte dell'organizzazione con ID `o-1234567890` per l'invio di log all'account del destinatario. Solo gli account con filtri di sottoscrizione creati invieranno effettivamente log all'account del destinatario.

Aggiornamento della policy di accesso alla destinazione nell'account del destinatario per iniziare a utilizzare un ID dell'organizzazione per le autorizzazioni

1. Nell'account del destinatario, utilizza un editor di testo per creare un file `~/AccessPolicy.json` con i seguenti contenuti.

```
{
    "Version" : "2012-10-17",
    "Statement" : [
        {
            "Sid" : "",
            "Effect" : "Allow",
            "Principal" : "*",
            "Action" : "logs:PutSubscriptionFilter",
            "Resource" :
 "arn:aws:logs:region:999999999999:destination:testDestination",
            "Condition": {
               "StringEquals" : {
                   "aws:PrincipalOrgID" : ["o-1234567890"]
                }
            }
        }
    ]
}
```

2. Digita il seguente comando per collegare la policy appena creata alla destinazione esistente. Per aggiornare una destinazione in modo da utilizzare una policy di accesso con un ID dell'organizzazione al posto di una policy di accesso che elenca ID account AWS specifici, includi il parametro `force`.

> **Warning**
>
> Se utilizzi i log inviati da un AWS servizio elencato in[Abilitazione della registrazione dai servizi AWS](), prima di eseguire questo passaggio devi aver aggiornato i filtri di

abbonamento in tutti gli account mittente, come spiegato in. [Fase 1: aggiornamento dei filtri di sottoscrizione]

```
aws logs put-destination-policy
    \ --destination-name "testDestination"
    \ --access-policy file://~/AccessPolicy.json
    \ --force
```

Condivisione di dati di log tra più account tramite Kinesis Data Firehose

Per condividere dati di log tra diversi account, è necessario stabilire un mittente e un ricevitore di dati di log:

- Registra il mittente dei dati: ottiene le informazioni sulla destinazione dal destinatario e comunica a CloudWatch Logs che è pronto a inviare gli eventi di registro alla destinazione specificata. Nelle procedure illustrate nel resto di questa sezione, il mittente dei dati di registro viene visualizzato con un numero di account fittizio pari a 1111 AWS .
- Destinatario dei dati di registro: imposta una destinazione che incapsula un flusso Kinesis Data Streams e CloudWatch fa sapere a Logs che il destinatario desidera ricevere i dati di registro. Il destinatario quindi condivide le informazioni su questa destinazione con il mittente. Nelle procedure descritte nel resto di questa sezione, il destinatario dei dati di registro viene mostrato con un numero di account fittizio di 222222222222. AWS

L'esempio in questa sezione utilizza un flusso di consegna Kinesis Data Firehose con archiviazione Amazon S3. È inoltre possibile configurare flussi di consegna Kinesis Data Firehose con impostazioni diverse. Per ulteriori informazioni, consulta [Creazione di un flusso di consegna Kinesis Data Firehose].

Il gruppo di log e la destinazione devono trovarsi nella stessa regione. AWS Tuttavia, la risorsa AWS a cui punta la destinazione può trovarsi in una Regione diversa.

> **Note**
>
> Il filtro di sottoscrizione per Kinesis Data Firehose per un flusso di consegna di uno stesso account e tra Regioni è supportato.

Argomenti

Fase 1: creazione di un flusso di consegna Kinesis Data Firehose

> **Important**
>
> Prima di completare i seguenti passaggi, è necessario utilizzare una policy di accesso, in modo che Kinesis Data Firehose possa accedere al bucket Amazon S3. Per ulteriori informazioni, consulta Controllo degli accessi nella Guida per sviluppatori di Amazon Kinesis Data Firehose.
> Tutti i passaggi in questa sezione (Fase 1) devono essere eseguiti nell'account del destinatario dei dati di log.
> La regione Stati Uniti orientali (Virginia settentrionale) viene utilizzata nei comandi di esempio. Sostituiscila con la Regione corretta per l'implementazione.

Per creare un flusso di consegna di Kinesis Data Firehose da utilizzare come destinazione.

1. Crea un bucket Amazon S3:

```
aws s3api create-bucket --bucket firehose-test-bucket1 --create-bucket-
configuration LocationConstraint=us-east-1
```

2. Crea il ruolo IAM che concede a Kinesis Data Firehose l'autorizzazione per inserire dati nel bucket.

   a. In primo luogo, utilizza un editor di testo per creare una policy di attendibilità in un file `~/TrustPolicyForFirehose.json`.

```
{ "Statement": { "Effect": "Allow", "Principal": { "Service":
 "firehose.amazonaws.com" }, "Action": "sts:AssumeRole", "Condition":
 { "StringEquals": { "sts:ExternalId":"222222222222" } } } }
```

b. Crea il ruolo IAM, specificando il file della policy di attendibilità appena creato.

```
aws iam create-role \
    --role-name FirehosetoS3Role \
    --assume-role-policy-document file://~/TrustPolicyForFirehose.json
```

c. L'output di questo comando risulterà simile al seguente: Annotare i Nome ruolo e ARN ruolo.

```
{
    "Role": {
        "Path": "/",
        "RoleName": "FirehosetoS3Role",
        "RoleId": "AROAR3BXASEKW7K635M53",
        "Arn": "arn:aws:iam::222222222222:role/FirehosetoS3Role",
        "CreateDate": "2021-02-02T07:53:10+00:00",
        "AssumeRolePolicyDocument": {
            "Statement": {
                "Effect": "Allow",
                "Principal": {
                    "Service": "firehose.amazonaws.com"
                },
                "Action": "sts:AssumeRole",
                "Condition": {
                    "StringEquals": {
                        "sts:ExternalId": "222222222222"
                    }
                }
            }
        }
    }
}
```

3. Crea una policy di autorizzazioni per definire le operazioni che Kinesis Data Firehose può eseguire nel tuo account.

a. In primo luogo, utilizza un editor di testo per creare la policy di autorizzazione seguente in un file denominato `~/PermissionsForFirehose.json`. A seconda del caso d'uso, potresti dover aggiungere altre autorizzazioni a questo file.

```
{
    "Statement": [{
        "Effect": "Allow",
        "Action": [
            "s3:PutObject",
            "s3:PutObjectAcl",
            "s3:ListBucket"
        ],
        "Resource": [
            "arn:aws:s3:::firehose-test-bucket1",
            "arn:aws:s3:::firehose-test-bucket1/*"
        ]
    }]
}
```

b. Inserisci il comando seguente per associare la policy di autorizzazioni appena creata con il ruolo IAM.

```
aws iam put-role-policy --role-name FirehosetoS3Role --policy-name
 Permissions-Policy-For-Firehose-To-S3 --policy-document file://~/
PermissionsForFirehose.json
```

4. Inserisci il comando seguente per creare il flusso di consegna Kinesis Data Firehose. Sostituisci *my-role-arn* e *my-bucket-arn* con i valori corretti per la tua distribuzione.

```
aws firehose create-delivery-stream \
   --delivery-stream-name 'my-delivery-stream' \
   --s3-destination-configuration \
  '{"RoleARN": "arn:aws:iam::222222222222:role/FirehosetoS3Role", "BucketARN":
 "arn:aws:s3:::firehose-test-bucket1"}'
```

L'output visualizzato dovrebbe essere simile al seguente:

```
{
    "DeliveryStreamARN": "arn:aws:firehose:us-east-1:222222222222:deliverystream/
my-delivery-stream"
}
```

Fase 2: creazione di una destinazione

> **Important**
>
> Tutte le fasi di questa procedura devono essere eseguite nell'account del destinatario dei dati di log.

Quando viene creata la destinazione, CloudWatch Logs invia un messaggio di prova alla destinazione per conto dell'account del destinatario. Quando il filtro di sottoscrizione è attivo in un secondo momento, CloudWatch Logs invia gli eventi di registro alla destinazione per conto dell'account di origine.

Creazione di una destinazione

1. Attendere fino a quando il flusso di Kinesis Data Firehose creato in Fase 1: creazione di un flusso di consegna Kinesis Data Firehose diventa attivo. È possibile utilizzare il seguente comando per controllare ilStreamDescription. StreamStatusproprietà.

   ```
   aws firehose describe-delivery-stream --delivery-stream-name "my-delivery-stream"
   ```

   Inoltre, prendi nota del DeliveryStreamDescription. DeliveryStreamValore ARN, perché sarà necessario utilizzarlo in un passaggio successivo. Esempio di output di questo comando:

   ```
   {
       "DeliveryStreamDescription": {
           "DeliveryStreamName": "my-delivery-stream",
           "DeliveryStreamARN": "arn:aws:firehose:us-
   east-1:222222222222:deliverystream/my-delivery-stream",
           "DeliveryStreamStatus": "ACTIVE",
           "DeliveryStreamEncryptionConfiguration": {
               "Status": "DISABLED"
           },
           "DeliveryStreamType": "DirectPut",
           "VersionId": "1",
           "CreateTimestamp": "2021-02-01T23:59:15.567000-08:00",
           "Destinations": [
               {
                   "DestinationId": "destinationId-000000000001",
                   "S3DestinationDescription": {
   ```

```
                    "RoleARN": "arn:aws:iam::222222222222:role/FirehosetoS3Role",
                    "BucketARN": "arn:aws:s3:::firehose-test-bucket1",
                    "BufferingHints": {
                        "SizeInMBs": 5,
                        "IntervalInSeconds": 300
                    },
                    "CompressionFormat": "UNCOMPRESSED",
                    "EncryptionConfiguration": {
                        "NoEncryptionConfig": "NoEncryption"
                    },
                    "CloudWatchLoggingOptions": {
                        "Enabled": false
                    }
                },
                "ExtendedS3DestinationDescription": {
                    "RoleARN": "arn:aws:iam::222222222222:role/FirehosetoS3Role",
                    "BucketARN": "arn:aws:s3:::firehose-test-bucket1",
                    "BufferingHints": {
                        "SizeInMBs": 5,
                        "IntervalInSeconds": 300
                    },
                    "CompressionFormat": "UNCOMPRESSED",
                    "EncryptionConfiguration": {
                        "NoEncryptionConfig": "NoEncryption"
                    },
                    "CloudWatchLoggingOptions": {
                        "Enabled": false
                    },
                    "S3BackupMode": "Disabled"
                }
            }
        ],
        "HasMoreDestinations": false
    }
}
```

Potrebbero essere necessari uno o due minuti che il flusso di consegna venga visualizzato nello stato attivo.

2. Quando il flusso di distribuzione è attivo, crea il ruolo IAM che concederà a CloudWatch Logs l'autorizzazione a inserire i dati nel tuo flusso Kinesis Data Firehose. Per prima cosa, devi creare una policy di fiducia in un file ~/ CWL.json. TrustPolicyFor Utilizza un editor di testo per creare

questa policy. Per ulteriori informazioni sugli endpoint di Amazon CloudWatch Logs, consulta Endpoints e quote di Amazon CloudWatch Logs.

Questa policy include una chiave di contesto della condizione globale `aws:SourceArn` che specifica il `sourceAccountId` per prevenire il problema di sicurezza noto come "confused deputy". Se non conosci ancora l'ID dell'account di origine nella prima chiamata, consigliamo di inserire l'ARN di destinazione nel campo ARN di origine. Nelle chiamate successive, è necessario impostare l'ARN di origine come l'ARN di origine effettivo raccolto dalla prima chiamata. Per ulteriori informazioni, consulta Prevenzione del "confused deputy".

```
{
    "Statement": {
        "Effect": "Allow",
        "Principal": {
            "Service": "logs.region.amazonaws.com"
        },
        "Action": "sts:AssumeRole",
        "Condition": {
            "StringLike": {
                "aws:SourceArn": [
                    "arn:aws:logs:region:sourceAccountId:*",
                    "arn:aws:logs:region:recipientAccountId:*"
                ]
            }
        }
     }
}
```

3. Utilizza il comando aws iam create-role per creare il ruolo IAM, specificando il file della policy di attendibilità appena creato.

```
aws iam create-role \
      --role-name CWLtoKinesisFirehoseRole \
      --assume-role-policy-document file://~/TrustPolicyForCWL.json
```

Di seguito è riportato un output di esempio. Prendi nota del valore `Role.Arn` restituito, perché dovrai usarlo in una fase successiva.

```
{
    "Role": {
        "Path": "/",
```

```
        "RoleName": "CWLtoKinesisFirehoseRole",
        "RoleId": "AROAR3BXASEKYJYWF243H",
        "Arn": "arn:aws:iam::222222222222:role/CWLtoKinesisFirehoseRole",
        "CreateDate": "2021-02-02T08:10:43+00:00",
        "AssumeRolePolicyDocument": {
            "Statement": {
                "Effect": "Allow",
                "Principal": {
                    "Service": "logs.region.amazonaws.com"
                },
                "Action": "sts:AssumeRole",
                "Condition": {
                    "StringLike": {
                        "aws:SourceArn": [
                            "arn:aws:logs:region:sourceAccountId:*",
                            "arn:aws:logs:region:recipientAccountId:*"
                        ]
                    }
                }
            }
        }
    }
}
```

4. Crea una politica di autorizzazioni per definire quali azioni CloudWatch Logs può eseguire sul tuo account. Innanzitutto, usa un editor di testo per creare una politica di autorizzazioni in un file ~/ CWL.json: PermissionsFor

```
{
    "Statement":[
      {
        "Effect":"Allow",
        "Action":["firehose:*"],
        "Resource":["arn:aws:firehose:region:222222222222:*"]
      }
    ]
}
```

5. Associa la policy di autorizzazioni al ruolo inserendo il comando seguente:

```
aws iam put-role-policy --role-name CWLtoKinesisFirehoseRole --policy-name
 Permissions-Policy-For-CWL --policy-document file://~/PermissionsForCWL.json
```

6. Dopo che il flusso di distribuzione di Kinesis Data Firehose è nello stato attivo e hai creato il ruolo IAM, puoi CloudWatch creare la destinazione Logs.

   a. In questa fase alla tua destinazione non verrà associata una policy d'accesso predefinita e costituisce solo la prima delle due fasi necessarie per completare la creazione della destinazione. Annota dell'ARN della nuova destinazione restituito nel payload, perché lo utilizzerai come `destination.arn` in una fase successiva.

```
aws logs put-destination \

    --destination-name "testFirehoseDestination" \
    --target-arn "arn:aws:firehose:us-east-1:222222222222:deliverystream/my-
delivery-stream" \
    --role-arn "arn:aws:iam::222222222222:role/CWLtoKinesisFirehoseRole"

{
    "destination": {
        "destinationName": "testFirehoseDestination",
        "targetArn": "arn:aws:firehose:us-east-1:222222222222:deliverystream/
my-delivery-stream",
        "roleArn": "arn:aws:iam::222222222222:role/CWLtoKinesisFirehoseRole",
        "arn": "arn:aws:logs:us-
east-1:222222222222:destination:testFirehoseDestination"}
}
```

   b. Dopo aver completato la fase precedente, nell'account del destinatario dei dati di log (222222222222) associa alla destinazione una policy d'accesso.

      Questa policy consente all'account mittente dei dati di log (111111111111) di accedere alla destinazione all'interno dell'account destinatario dei dati di log (222222222222). Puoi usare un editor di testo per inserire questa policy nel file AccessPolicy~/ .json:

```
{
  "Version" : "2012-10-17",
  "Statement" : [
    {
      "Sid" : "",
      "Effect" : "Allow",
      "Principal" : {
        "AWS" : "111111111111"
      },
      "Action" : "logs:PutSubscriptionFilter",
```

```
      "Resource" : "arn:aws:logs:us-
east-1:222222222222:destination:testFirehoseDestination"
    }
  ]
}
```

c. In questo modo viene creata una policy che definisce chi ha accesso in scrittura alla destinazione. Questa politica deve specificare i log: PutSubscriptionFilter action per accedere alla destinazione. Gli utenti con più account utilizzeranno l'PutSubscriptionFilterazione per inviare gli eventi di registro alla destinazione:

```
aws logs put-destination-policy \
    --destination-name "testFirehoseDestination" \
    --access-policy file://~/AccessPolicy.json
```

Passaggio 3: aggiunta/convalida delle autorizzazioni IAM per la destinazione multi-account

In base alla AWS logica di valutazione dei criteri tra account, per accedere a qualsiasi risorsa tra account (ad esempio uno stream Kinesis o Kinesis Data Firehose utilizzato come destinazione per un filtro di sottoscrizione) è necessario disporre di una politica basata sull'identità nell'account di invio che fornisca l'accesso esplicito alla risorsa di destinazione tra account diversi. Per ulteriori informazioni sulla logica di valutazione delle policy, consulta la pagina Cross-account policy evaluation logic.

Puoi collegare la policy basata sull'identità al ruolo IAM o all'utente IAM che stai utilizzando per creare il filtro di sottoscrizione. Questa policy deve essere presente nell'account mittente. Se utilizzi il ruolo di amministratore per creare il filtro di sottoscrizione, puoi saltare questo passaggio e passare a Passaggio 4: creazione di un filtro di sottoscrizione.

Aggiunta o convalida delle autorizzazioni IAM necessarie per più account

1. Inserisci il seguente comando per verificare quale ruolo IAM o utente IAM viene utilizzato per eseguire i comandi di log AWS .

   ```
   aws sts get-caller-identity
   ```

   Il comando restituisce un output simile al seguente:

```
{
"UserId": "User ID",
"Account": "sending account id",
"Arn": "arn:aws:sending account id:role/user:RoleName/UserName"
}
```

Prendi nota del valore rappresentato da o. *RoleNameUserName*

2. AWS Management Console Accedi all'account di invio e cerca le policy allegate con il ruolo IAM o l'utente IAM restituito nell'output del comando che hai inserito nel passaggio 1.
3. Verifica che le policy associate a questo ruolo o utente forniscano autorizzazioni esplicite per richiamare `logs:putSubscriptionFilter` sulla risorsa di destinazione multi-account. Le seguenti policy di esempio mostrano le autorizzazioni suggerite.

   La seguente politica fornisce le autorizzazioni per creare un filtro di sottoscrizione su qualsiasi risorsa di destinazione solo in un singolo AWS account, account`123456789012`:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Allow subscription filters on any resource in one specific
 account",
            "Effect": "Allow",
            "Action": "logs:PutSubscriptionFilter",
            "Resource": [
                "arn:aws:logs:*:*:log-group:*",
                "arn:aws:logs:*:123456789012:destination:*"
            ]
        }
    ]
}
```

   La seguente politica fornisce le autorizzazioni per creare un filtro di sottoscrizione solo su una risorsa di destinazione specifica denominata `sampleDestination` in AWS account singolo, account: `123456789012`

```
{
    "Version": "2012-10-17",
    "Statement": [
```

```
        {
            "Sid": "Allow subscription filters on one specific resource in one
 specific account",
            "Effect": "Allow",
            "Action": "logs:PutSubscriptionFilter",
            "Resource": [
                "arn:aws:logs:*:*:log-group:*",
                "arn:aws:logs:*:123456789012:destination:sampleDestination"
            ]
        }
    ]
}
```

Passaggio 4: creazione di un filtro di sottoscrizione

Passare all'account di invio, che in questo esempio è 111111111111. Verrà ora creato il filtro di sottoscrizione nell'account di invio. In questo esempio, il filtro è associato a un gruppo di log contenente AWS CloudTrail eventi in modo che ogni attività registrata effettuata dalle AWS credenziali «Root» venga consegnata alla destinazione creata in precedenza. Per ulteriori informazioni su come inviare AWS CloudTrail eventi ai CloudWatch registri, vedere Invio di CloudTrail eventi ai CloudWatch registri nella Guida per l'utente.AWS CloudTrail

Quando inserisci il comando seguente, assicurati di aver effettuato l'accesso come utente IAM o di utilizzare il ruolo IAM per cui hai aggiunto la policy in Passaggio 3: aggiunta/convalida delle autorizzazioni IAM per la destinazione multi-account.

```
aws logs put-subscription-filter \
    --log-group-name "aws-cloudtrail-logs-111111111111-300a971e" \
    --filter-name "firehose_test" \
    --filter-pattern "{$.userIdentity.type = AssumedRole}" \
    --destination-arn "arn:aws:logs:us-
east-1:222222222222:destination:testFirehoseDestination"
```

Il gruppo di log e la destinazione devono trovarsi nella stessa AWS regione. Tuttavia, la destinazione può puntare a una AWS risorsa come un flusso Kinesis Data Firehose che si trova in una regione diversa.

Convalida del flusso dei log eventi

Dopo aver creato il filtro di sottoscrizione, CloudWatch Logs inoltra tutti gli eventi di registro in entrata che corrispondono allo schema di filtro al flusso di distribuzione di Kinesis Data Firehose. I dati iniziano a comparire nel bucket Amazon S3 in base all'intervallo del buffer di tempo impostato nel flusso di consegna Kinesis Data Firehose. Quando è trascorso tempo sufficiente, puoi verificare i dati controllando il bucket Amazon S3. Per controllare il bucket, inserisci il comando seguente:

```
aws s3api list-objects --bucket 'firehose-test-bucket1'
```

L'output di questo comando sarà simile al seguente:

```
{
    "Contents": [
        {
            "Key": "2021/02/02/08/my-delivery-
stream-1-2021-02-02-08-55-24-5e6dc317-071b-45ba-a9d3-4805ba39c2ba",
            "LastModified": "2021-02-02T09:00:26+00:00",
            "ETag": "\"EXAMPLEa817fb88fc770b81c8f990d\"",
            "Size": 198,
            "StorageClass": "STANDARD",
            "Owner": {
                "DisplayName": "firehose+2test",
                "ID": "EXAMPLE27fd05889c665d2636218451970ef79400e3d2aecca3adb1930042e0"
            }
        }
    ]
}
```

È quindi possibile recuperare un oggetto specifico dal bucket inserendo il seguente comando. Sostituisci il valore di `key` con il valore trovato nel comando precedente.

```
aws s3api get-object --bucket 'firehose-test-bucket1' --key '2021/02/02/08/my-delivery-
stream-1-2021-02-02-08-55-24-5e6dc317-071b-45ba-a9d3-4805ba39c2ba' testfile.gz
```

I dati nell'oggetto di Amazon S3 vengono compressi nel formato gzip. Puoi esaminare i dati non elaborati dalla riga di comando utilizzando uno dei seguenti comandi:

Linux:

```
zcat testfile.gz
```

macOS:

```
zcat <testfile.gz
```

Modifica dell'appartenenza alla destinazione durante il runtime

Potrebbero verificarsi situazioni in cui devi aggiungere o rimuovere mittenti di log da una destinazione da te posseduta. Puoi utilizzare l'PutDestinationPolicyazione sulla tua destinazione con una nuova politica di accesso. Nell'esempio seguente, si impedisce a un account aggiunto in precedenza 111111111111 di inviare ulteriori dati di log, mentre l'account 333333333333 viene abilitato.

1. Recupera la politica attualmente associata alla destinazione TestDestination e prendi nota di: AccessPolicy

```
aws logs describe-destinations \
    --destination-name-prefix "testFirehoseDestination"

{
    "destinations": [
        {
            "destinationName": "testFirehoseDestination",
            "targetArn": "arn:aws:firehose:us-east-1:222222222222:deliverystream/
my-delivery-stream",
            "roleArn": "arn:aws:iam:: 222222222222:role/CWLtoKinesisFirehoseRole",
            "accessPolicy": "{\n  \"Version\" : \"2012-10-17\",\n  \"Statement
\" : [\n    {\n      \"Sid\" : \"\",\n      \"Effect\" : \"Allow\",\n
 \"Principal\" : {\n        \"AWS\" : \"111111111111 \"\n      },\n      \"Action
\" : \"logs:PutSubscriptionFilter\",\n      \"Resource\" : \"arn:aws:logs:us-
east-1:222222222222:destination:testFirehoseDestination\"\n    }\n  ]\n}\n\n",
            "arn": "arn:aws:logs:us-east-1:
 222222222222:destination:testFirehoseDestination",
            "creationTime": 1612256124430
        }
    ]
}
```

2. Aggiorna la policy per riflettere che l'account 111111111111 è stoppato, mentre l'account 333333333333 viene abilitato. Inserisci questa politica nel file ~/ .json: NewAccessPolicy

```
{
  "Version" : "2012-10-17",
  "Statement" : [
```

```
    {
      "Sid" : "",
      "Effect" : "Allow",
      "Principal" : {
        "AWS" : "333333333333 "
      },
      "Action" : "logs:PutSubscriptionFilter",
      "Resource" : "arn:aws:logs:us-
east-1:222222222222:destination:testFirehoseDestination"
    }
  ]
}
```

3. Usa il seguente comando per associare la politica definita nel NewAccessPolicyfile.json alla destinazione:

```
aws logs put-destination-policy \
    --destination-name "testFirehoseDestination" \

    --access-policy file://~/NewAccessPolicy.json
```

Questo alla fine disabilita il log eventi dall'ID account 111111111111. I log eventi provenienti dall'ID account 333333333333 iniziano a fluire verso la destinazione non appena il proprietario dell'account 333333333333 crea un filtro di sottoscrizione.

Prevenzione del "confused deputy"

Con "confused deputy" si intende un problema di sicurezza in cui un'entità che non dispone dell'autorizzazione per eseguire una certa operazione può costringere un'entità con più privilegi a eseguire tale operazione. Nel AWS, l'impersonificazione tra servizi può portare al confuso problema del vice. La rappresentazione tra servizi può verificarsi quando un servizio (il servizio chiamante) effettua una chiamata a un altro servizio (il servizio chiamato). Il servizio chiamante può essere manipolato per utilizzare le proprie autorizzazioni e agire sulle risorse di un altro cliente, a cui normalmente non avrebbe accesso. Per evitare che ciò accada, AWS mette a disposizione strumenti che consentono di proteggere i dati relativi a tutti i servizi, con responsabili del servizio a cui è stato concesso l'accesso alle risorse del vostro account.

Ti consigliamo di utilizzare le chiavi di contesto `aws:SourceArn` o `aws:SourceAccount` global condition nelle politiche delle risorse per limitare l'ambito delle autorizzazioni concesse ai CloudWatch Logs per scrivere dati su Kinesis Data Streams e Kinesis Data Firehose.

Il valore di `aws:SourceArn` deve limitare le autorizzazioni ai soli account che scrivono e ricevono dati.

Il modo più efficace per proteggersi dal problema "confused deputy" è quello di usare la chiave di contesto della condizione globale `aws:SourceArn` con l'ARN completo della risorsa. Se non conosci l'ARN completo della risorsa o se stai specificando più risorse, usa la chiave aws: SourceArn global context condition with wildcards (*) per le parti sconosciute dell'ARN. Ad esempio, `arn:aws:servicename::123456789012:*`.

Le politiche documentate per concedere l'accesso ai CloudWatch log per scrivere dati su Kinesis Data Streams e Kinesis Data Firehose mostrano come utilizzare la chiave SourceArn aws: global condition context per aiutare Fase 2: creazione di una destinazione a prevenire il confuso problema del vice. Passaggio 1: creazione di una destinazione

Sintassi del modello di filtro per filtri di parametri, filtri di sottoscrizione, filtri di log eventi e Live Tail

> **Note**
>
> Per informazioni su come eseguire query sui gruppi di log con il linguaggio di query di Logs Insights di Amazon CloudWatch, consulta la pagina CloudWatch Sintassi delle query di Logs Insights.

Con File di log CloudWatch, puoi utilizzare filtri di parametri per trasformare i dati di log in parametri utilizzabili, filtri di sottoscrizione per indirizzare i log eventi ad altri servizi AWS, filtri di log eventi per cercare eventi di log e Live Tail per visualizzare in modo interattivo i log in tempo reale man mano che vengono importati.

I modelli di filtro costituiscono la sintassi utilizzata dai filtri di parametri, dai filtri di sottoscrizione, dai filtri di log eventi e da Live Tail per verificare la presenza di termini nei log eventi. I termini possono essere parole, frasi esatte o valori numerici. Le espressioni regolari (regex) possono essere utilizzate per creare modelli di filtro autonomi o incorporate con modelli di filtri JSON delimitati da spazi.

Crea modelli di filtro con i termini che desideri associare. I modelli di filtro restituiscono solo i log eventi che contengono i termini definiti. Puoi testare i modelli di filtro nella console CloudWatch.

Argomenti

Sintassi delle espressioni regolari (regex) supportate

Sintassi regex supportata

Quando si utilizzano espressioni regolari per cercare e filtrare i dati di un log, è necessario racchiudere le espressioni tra %.

I modelli di filtro con regex possono includere solo quanto segue:

- Caratteri alfanumerici: un carattere alfanumerico è un carattere costituito da una lettera (dalla A alla Z o dalla a alla z) o una cifra (da 0 a 9).
- Caratteri simbolici supportati: '_', '#', '=', '@','/', ';', ',' e '-'. Ad esempio, `%something!%` verrebbe rifiutato poiché '!' non è supportato.
- Operatori supportati: questi includono: '^', '$', '?', '[', ']', '{', '}', '|', '\', '*', '+' e '.'.

Gli operatori (e) non sono supportati. Non è possibile utilizzare le parentesi per definire un sottomodello.

I caratteri multibyte non sono supportati.

> **Note**
>
> Quote
> Quando si creano filtri di parametri o filtri di sottoscrizione, per ogni gruppo di log sono disponibili al massimo 5 modelli di filtro contenenti espressioni regolari.
> Quando si crea un modello di filtro delimitato o JSON per filtri di parametri e filtri di sottoscrizione o quando si filtrano log eventi, è possibile utilizzare al massimo due espressioni regolari per ogni modello di filtro.

Utilizzo degli operatori supportati

- ^: fissa la corrispondenza all'inizio di una stringa. Ad esempio, `%^[hc]at%` restituisce "hat" e "cat", ma solo all'inizio di una stringa.
- $: fissa la corrispondenza all'inizio di una stringa. Ad esempio, `%[hc]at$%` corrisponde a "hat" e "cat", ma solo alla fine di una stringa.
- ?: restituisce zero o più istanze del termine precedente. Ad esempio, `%colou?r%` può restituire entrambe le varianti "color" e "colour".

- `[]`: definisce una classe di caratteri. Corrisponde all'elenco o all'intervallo di caratteri racchiuso tra parentesi quadre. Ad esempio, `%[abc]%` restituisce ad "a", "b" o "c", `%[a-z]%` restituisce qualsiasi lettera minuscola da "a" a "z" e `%[abcx-z]%` restituisce ad "a", "b", "c", "x", "y" oppure "z".
- `{m, n}`: restituisce il termine precedente almeno m e non più di n volte. Ad esempio, `%a{3,5}%` restituisce solo "aaa", "aaaa" e "aaaaa".

> **Note**
>
> È possibile omettere m o n se si sceglie di non definire un minimo o un massimo.

- `|`: "or" booleano, che restituisce il termine a sinistra o a destra della barra verticale. Ad esempio, `%gra|ey%` può restituire entrambe le varianti "gray" o "grey" (grigio).

> **Note**
>
> Un termine è un carattere singolo o una classe di caratteri ripetuti che utilizza uno dei seguenti operatori: `?`, `*`, `+` o `{n,m}`.

- `\`: carattere di escape, che consente di utilizzare il significato letterale di un operatore anziché il suo significato speciale. Ad esempio, `%\[.\]%` restituisce qualsiasi carattere singolo racchiuso tra "[" e "]" poiché le parentesi quadre sono ignorate, come nel caso di "[a]", "[b]", "[7]", "[@]", "[]]" e "[ ]".

> **Note**
>
> `%10\.10\.0\.1%` è il modo corretto per creare un'espressione regolare che restituisca l'indirizzo IP 10.10.0.1.

- `*`: restituisce zero o più istanze del termine precedente. Ad esempio, `%ab*c%` può restituire "ac", "abc" e "abbbc"; `%ab[0-9]*%` può restituire "ab", "ab0" e "ab129".
- `+`: corrisponde a una o più istanze del termine precedente. Ad esempio, `%ab+c%` può restituire "abc", "abbc" e "abbbc", ma non "ac".
- `.`: corrisponde a qualsiasi carattere singolo. Ad esempio, `%.at%` restituisce qualsiasi stringa di tre caratteri che termina con "at", tra cui "hat", "cat", "bat", "4at", "#at" e " at" (che inizia con uno spazio).

> **Note**
>
> Quando si crea un'espressione regolare per verificare la presenza di indirizzi IP, è importante evitare l'operatore `.`. Ad esempio, `%10.10.0.1%` può restituire "10010,051", che potrebbe non essere lo scopo effettivo previsto dell'espressione.

- `\d`, `\D`: restituisce un carattere numerico/non numerico. Ad esempio, `%\d%` è equivalente a `%[0-9]%` e `%\D%` è equivalente a `%[^0-9]%`.

> **Note**
>
> L'operatore maiuscolo indica l'inverso della sua controparte minuscola.

- `\s`, `\S`: restituisce un carattere di spazio bianco/carattere diverso da uno spazio bianco.

> **Note**
>
> L'operatore maiuscolo indica l'inverso della sua controparte minuscola. I caratteri di spaziatura includono i caratteri tab (`\t`), spazio ( ) e newline (`\n`).

- `\w`, `\W`: restituisce un carattere alfanumerico/non alfanumerico. Ad esempio, `%\w%` è equivalente a `%[a-zA-Z_0-9]%` e `%\W%` è equivalente a `%[^a-zA-Z_0-9]%`.

> **Note**
>
> L'operatore maiuscolo indica l'inverso della sua controparte minuscola.

- `\xhh`: restituisce la mappatura ASCII per un carattere esadecimale a due cifre. `\x` è la sequenza di escape che indica che i caratteri seguenti rappresentano il valore esadecimale per ASCII. `hh` specifica le due cifre esadecimali (0-9 e A-F) che puntano a un carattere nella tabella ASCII.

> **Note**
>
> È possibile utilizzare `\xhh` per restituire caratteri simbolici che non sono supportati dal modello di filtro. Ad esempio, `%\x3A%` corrisponde a `:` e `%\x28%` corrisponde a `(`.

Utilizzo dei modelli di filtro per verificare la presenza dei termini in un'espressione regolare (regex)

Corrispondenza dei termini usando regex

È possibile verificare la presenza di termini nei log eventi utilizzando un modello di espressioni regolari racchiuso tra % (segni percentuali prima e dopo il modello di espressione regolare). Il seguente frammento di codice mostra un esempio di un modello di filtro che restituisce tutti i log eventi che comprendono la parola chiave AUTHORIZED.

Per un elenco delle espressioni regolari supportate, consulta la sezione Espressioni regolari supportate.

```
%AUTHORIZED%
```

Questo modello di filtro restituisce messaggi di log eventi come i seguenti:

- `[ERROR 401] UNAUTHORIZED REQUEST`
- `[SUCCESS 200] AUTHORIZED REQUEST`

Utilizzo dei modelli di filtro per associare termini nei log eventi non strutturati

Corrispondenza dei termini in log eventi non strutturati

Gli esempi seguenti contengono frammenti di codice che mostrano come è possibile utilizzare i modelli di filtro per associare i termini nei log eventi non strutturati.

> **Note**
>
> I modelli di filtro fanno distinzione tra maiuscole e minuscole. Racchiudi frasi e termini esatti che includono caratteri non alfanumerici tra virgolette doppie ("").

Example: Match a single term

Il seguente frammento di codice mostra un esempio del modello di filtro di un termine singolo che restituisce tutti i log eventi in cui i messaggi contengono la parola ERROR.

```
ERROR
```

Questo modello di filtro corrisponde ai messaggi di log eventi come i seguenti:

- `[ERROR 400] BAD REQUEST`
- `[ERROR 401] UNAUTHORIZED REQUEST`
- `[ERROR 419] MISSING ARGUMENTS`
- `[ERROR 420] INVALID ARGUMENTS`

Example: Match multiple terms

Il seguente frammento di codice mostra un esempio del modello di filtro di più termini che restituisce tutti i log eventi in cui i messaggi contengono le parole ERROR e ARGUMENTS.

```
ERROR ARGUMENTS
```

Il filtro restituisce i messaggi del log eventi, come i seguenti:

- `[ERROR 419] MISSING ARGUMENTS`
- `[ERROR 420] INVALID ARGUMENTS`

Questo modello di filtro non restituisce i seguenti messaggi del log eventi perché non contengono entrambi i termini specificati nel modello di filtro.

- `[ERROR 400] BAD REQUEST`
- `[ERROR 401] UNAUTHORIZED REQUEST`

Example: Match optional terms

È possibile utilizzare la corrispondenza di modelli per creare modelli di filtro che restituiscono log eventi contenenti termini facoltativi. Inserisci un punto interrogativo ("?") prima dei termini che desideri associare. Il seguente frammento di codice mostra un esempio di un modello di filtro che restituisce tutti i log eventi in cui i messaggi contengono la parola ERRORE o ARGOMENTI.

```
?ERROR ?ARGUMENTS
```

Questo modello di filtro corrisponde ai messaggi di log eventi come i seguenti:

- `[ERROR 400] BAD REQUEST`
- `[ERROR 401] UNAUTHORIZED REQUEST`
- `[ERROR 419] MISSING ARGUMENTS`
- `[ERROR 420] INVALID ARGUMENTS`

> **Note**
>
> Non puoi combinare il punto interrogativo ("?") con altri modelli di filtro, come termini di inclusione ed esclusione. Se combini "?" con altri modelli di filtro, il punto interrogativo ("?") viene ignorato.
> Ad esempio, il seguente schema di filtro corrisponde a tutti gli eventi che contengono la parola `REQUEST`, ma il punto interrogativo ("?") il filtro viene ignorato e non ha alcun effetto.
>
> ```
> ?ERROR ?ARGUMENTS REQUEST
> ```
>
> Corrispondenze di log eventi
>
> - `[INFO] REQUEST FAILED`
> - `[WARN] UNAUTHORIZED REQUEST`
> - `[ERROR] 400 BAD REQUEST`

Example: Match exact phrases

Il seguente frammento di codice mostra un esempio di un modello di filtro che restituisce log eventi in cui i messaggi contengono la frase esatta INTERNAL SERVER ERROR.

```
"INTERNAL SERVER ERROR"
```

Questo modello di filtro restituisce il seguente messaggio di log eventi:

- `[ERROR 500] INTERNAL SERVER ERROR`

Example: Include and exclude terms

Puoi creare modelli di filtro che restituiscono log eventi in cui i messaggi includono alcuni termini ed escludono altri termini. Posiziona un simbolo meno ("-") prima dei termini che desideri escludere. Il seguente frammento di codice mostra un esempio di un modello di filtro che restituisce log eventi in cui i messaggi contengono il termine ERROR ed escludono il termine ARGUMENTS.

```
ERROR -ARGUMENTS
```

Questo modello di filtro restituisce messaggi di log eventi come i seguenti:

- `[ERROR 400] BAD REQUEST`
- `[ERROR 401] UNAUTHORIZED REQUEST`

Questo modello di filtro non restituisce i seguenti messaggi di log eventi perché contengono la parola ARGUMENTS.

- `[ERROR 419] MISSING ARGUMENTS`
- `[ERROR 420] INVALID ARGUMENTS`

Example: Match everything

Puoi utilizzare la corrispondenza totale nei log eventi usando le virgolette doppie. Il seguente frammento di codice mostra un esempio di un modello di filtro che restituisce tutti i log eventi.

```
" "
```

Utilizzo dei modelli di filtro per associare termini nei log eventi

Scrittura di modelli di filtro per log eventi JSON

Negli esempi seguenti viene descritto l'utilizzo della sintassi per i filtri di modelli che verificano la presenza di termini JSON contenenti stringhe e valori numerici.

Writing filter patterns that match strings

È possibile creare modelli di filtro per associare le stringhe nei log eventi JSON. Il seguente frammento di codice mostra un esempio della sintassi per il modello di filtro basato su stringa.

```
{ PropertySelector EqualityOperator String }
```

Racchiudi i modelli di filtro tra parentesi graffe ("{}"). I modelli di filtro basati su stringhe devono contenere le seguenti parti:

- Property selector (Selettore di proprietà)

  Imposta i selettori di proprietà con il simbolo del dollaro seguito da un punto ("$."). I selettori di proprietà sono stringhe alfanumeriche che supportano anche i caratteri trattino alto ("-") e basso ("_"). Le stringhe non supportano la notazione scientifica. I selettori di proprietà puntano ai nodi di valore nei log eventi JSON. I nodi di valore possono essere stringhe o numeri. Posiziona le matrici dopo i selettori di proprietà. Gli elementi negli array seguono un sistema di numerazione a base zero, il che significa che il primo elemento dell'array è l'elemento 0, il secondo elemento è l'elemento 1 e così via. Racchiudi gli elementi tra parentesi quadre ("[]"). Se un selettore di proprietà punta a una matrice o a un oggetto, il modello di filtro non corrisponde al formato

di log. Se la proprietà JSON contiene un punto (''.''), è possibile utilizzare la notazione a parentesi per selezionarla.

> **Note**
>
> Selettore di carattere jolly
> È possibile utilizzare il carattere jolly JSON per selezionare qualsiasi elemento dell'array o qualsiasi campo oggetto JSON.
> Quote
> È possibile utilizzare un solo selettore di caratteri jolly in un selettore di proprietà.

- Equality operator (Operatori di uguaglianza)

  Imposta gli operatori di uguaglianza con uno dei seguenti simboli: uguale ("=") o non uguale ("!=''). Gli operatori di uguaglianza restituiscono un valore booleano (vero o falso).

- String

  Puoi racchiudere le stringhe tra virgolette doppie (""). Le stringhe che non contengono caratteri alfanumerici e il simbolo del trattino basso devono essere collocate tra virgolette doppie. Usa l'asterisco ("*") come carattere jolly per associare il testo.

> **Note**
>
> È possibile utilizzare qualsiasi espressione regolare condizionale durante la creazione di modelli di filtro che verifichino la presenza di termini nei log eventi JSON. Per un elenco delle espressioni regolari supportate, consulta la sezione Espressioni regolari supportate.

Il seguente frammento di codice contiene un esempio di un modello di filtro che mostra come è possibile formattarne uno in modo che corrisponda a un termine JSON con una stringa.

```
{ $.eventType = "UpdateTrail" }
```

Writing filter patterns that match numeric values

È possibile creare modelli di filtro per associare valori numerici nei log eventi JSON. Il seguente frammento di codice mostra un esempio della sintassi per i modelli di filtro che corrispondono a valori numerici.

```
{ PropertySelector NumericOperator Number }
```

Racchiudi i modelli di filtro tra parentesi graffe ("{}"). I modelli di filtro che corrispondono a valori numerici devono avere le seguenti parti:

- Property selector (Selettore di proprietà)

  Imposta i selettori di proprietà con il simbolo del dollaro seguito da un punto ("$."). I selettori di proprietà sono stringhe alfanumeriche che supportano anche i caratteri trattino alto ("-") e basso ("_"). Le stringhe non supportano la notazione scientifica. I selettori di proprietà puntano ai nodi di valore nei log eventi JSON. I nodi di valore possono essere stringhe o numeri. Posiziona le matrici dopo i selettori di proprietà. Gli elementi negli array seguono un sistema di numerazione a base zero, il che significa che il primo elemento dell'array è l'elemento 0, il secondo elemento è l'elemento 1 e così via. Racchiudi gli elementi tra parentesi quadre ("[]"). Se un selettore di proprietà punta a una matrice o a un oggetto, il modello di filtro non corrisponde al formato di log. Se la proprietà JSON contiene un punto (" **.** "), è possibile utilizzare la notazione a parentesi per selezionarla.

  > **Note**
  >
  > Selettore di carattere jolly
  > È possibile utilizzare il carattere jolly JSON per selezionare qualsiasi elemento dell'array o qualsiasi campo oggetto JSON.
  > Quote
  > È possibile utilizzare un solo selettore di caratteri jolly in un selettore di proprietà.

- Operazioni numeriche

  Imposta le operazioni numeriche con uno dei seguenti simboli: maggiore di (">"), minore di ("<"), uguale ("="), non uguale ("!="), maggiore o uguale a (">=") oppure minore o uguale a ("<=").

- Numero

  Puoi utilizzare numeri interi che contengono i simboli più ("+") o meno ("-") e seguire la notazione scientifica. Usa l'asterisco ("*") come carattere jolly per associare i numeri.

Il seguente frammento di codice contiene esempi che mostrano come è possibile formattare i modelli di filtro in modo che corrispondano a termini JSON con valori numerici.

```
// Filter pattern with greater than symbol
{ $.bandwidth > 75 }
// Filter pattern with less than symbol
{ $.latency < 50 }
// Filter pattern with greater than or equal to symbol
{ $.refreshRate >= 60 }
// Filter pattern with less than or equal to symbol
{ $.responseTime <= 5 }
// Filter pattern with equal sign
{ $.errorCode = 400}
// Filter pattern with not equal sign
{ $.errorCode != 500 }
// Filter pattern with scientific notation and plus symbol
{ $.number[0] = 1e-3 }
// Filter pattern with scientific notation and minus symbol
{ $.number[0] != 1e+3 }
```

Corrispondenza dei termini in log eventi JSON utilizzando espressioni semplici

Gli esempi seguenti contengono frammenti di codice che mostrano come i modelli di filtro possono corrispondere ai termini in un log eventi JSON.

> **Note**
>
> Se testi il modello di filtro di esempio con il log eventi JSON di esempio, devi inserire il log JSON di esempio su una singola riga.

Log eventi JSON

```
{
      "eventType": "UpdateTrail",
```

```
      "sourceIPAddress": "111.111.111.111",
      "arrayKey": [
            "value",
            "another value"
      ],
      "objectList": [
           {
             "name": "a",
             "id": 1
           },
           {
             "name": "b",
             "id": 2
           }
      ],
      "SomeObject": null,
      "cluster.name": "c"
}
```

Example: Filter pattern that matches string values

Questo modello di filtro restituisce la stringa `"UpdateTrail"` nella proprietà `"eventType"`.

```
{ $.eventType = "UpdateTrail" }
```

Example: Filter pattern that matches string values (IP address)

Questo modello di filtro contiene un carattere jolly e corrisponde alla proprietà `"sourceIPAddress"` perché non contiene un numero con il prefisso `"123.123."`.

```
{ $.sourceIPAddress != 123.123.* }
```

Example: Filter pattern that matches a specific array element with a string value

Questo modello di filtro restituisce l'elemento `"value"` nell'array `"arrayKey"`.

```
{ $.arrayKey[0] = "value" }
```

Example: Filter pattern that matches a string using regex

Questo modello di filtro restituisce la stringa `"Trail"` nella proprietà `"eventType"`.

```
{ $.eventType = %Trail% }
```

Example: Filter pattern that uses a wildcard to match values of any element in the array using regex

Questo modello di filtro contiene espressioni regolari che restituiscono l'elemento `"value"` dell'array `"arrayKey"`.

```
{ $.arrayKey[*] = %val.{2}% }
```

Example: Filter pattern that uses a wildcard to match values of any element with a specific prefix and subnet using regex (IP address)

Questo modello di filtro contiene espressioni regolari che restituiscono l'elemento `"111.111.111.111"` della proprietà `"sourceIPAddress"`.

```
{ $.* = %111\.111\.111\.1[0-9]{1,2}% }
```

> **Note**
>
> Quote
> È possibile utilizzare un solo selettore di caratteri jolly in un selettore di proprietà.

Example: Filter pattern that matches a JSON property with a period (.) in the key

```
{ $.['cluster.name'] = "c" }
```

Example: Filter pattern that matches JSON logs using IS

È possibile creare modelli di filtro che restituiscono le corrispondenze con i campi dei log JSON con la variabile `IS`. La variabile `IS` può corrispondere a campi che contengono i valori `NULL`, `TRUE` oppure `FALSE`. Il seguente modello di filtro restituisce i log JSON in cui il valore di `SomeObject` è `NULL`.

```
{ $.SomeObject IS NULL }
```

Example: Filter pattern that matches JSON logs using NOT EXISTS

È possibile creare modelli di filtro con la variabile `NOT EXISTS` per restituire i log JSON che non contengono campi specifici nei dati di log. Il seguente modello di filtro utilizza `NOT EXISTS` per restituire i log JSON che non contengono il campo `SomeOtherObject`.

```
{ $.SomeOtherObject NOT EXISTS }
```

> **Note**
>
> Le variabili `IS NOT` e `EXISTS` al momento non sono supportate.

Corrispondenza di termini negli oggetti JSON utilizzando espressioni composte

È possibile utilizzare gli operatori logici AND ("&&") e OR ("||") nei modelli di filtro per creare espressioni composte che corrispondono ai log eventi in cui due o più condizioni sono vere. Le espressioni composte supportano l'uso di parentesi ("()") e il seguente ordine standard di operazioni: () > && > ||. Gli esempi seguenti contengono frammenti di codice che mostrano come utilizzare i modelli di filtro con le espressioni composte in modo da associare i termini in un oggetto JSON.

Oggetto JSON

```
{
    "user": {
        "id": 1,
        "email": "John.Stiles@example.com"
```

```
    },
    "users": [
        {
         "id": 2,
         "email": "John.Doe@example.com"
        },
        {
         "id": 3,
         "email": "Jane.Doe@example.com"
        }
    ],
    "actions": [
        "GET",
        "PUT",
        "DELETE"
    ],
    "coordinates": [
        [0, 1, 2],
        [4, 5, 6],
        [7, 8, 9]
    ]
}
```

Example: Expression that matches using AND (&&)

Questo modello di filtro contiene un'espressione composta che corrisponde a `"id"` in `"user"` con un valore numerico di `1` e `"email"` nel primo elemento dell'array `"users"` con la stringa `"John.Doe@example.com"`.

```
{ ($.user.id = 1) && ($.users[0].email = "John.Doe@example.com") }
```

Example: Expression that matches using OR (||)

Questo modello di filtro contiene un'espressione composta che corrisponde a `"email"` in `"user"` con la stringa `"John.Stiles@example.com"`.

```
{ $.user.email = "John.Stiles@example.com" || $.coordinates[0][1] = "nonmatch" &&
 $.actions[2] = "nonmatch" }
```

Example: Expression that doesn't match using AND (&&)

Questo modello di filtro contiene un'espressione composta che non trova una corrispondenza perché l'espressione non corrisponde alla terza operazione in `"actions"`.

```
{ ($.user.email = "John.Stiles@example.com" || $.coordinates[0][1] = "nonmatch") &&
 $.actions[2] = "nonmatch" }
```

> **Note**
>
> Quote
> È possibile utilizzare un solo selettore di caratteri jolly in un selettore di proprietà e fino a tre selettori di caratteri jolly in un modello di filtro con espressioni composte.

Example: Expression that doesn't match using OR (||)

Questo modello di filtro contiene un'espressione composta che non trova una corrispondenza perché l'espressione non corrisponde alla prima proprietà in `"users"` o alla terza operazione in `"actions"`.

```
{ ($.user.id = 2 && $.users[0].email = "nonmatch") || $.actions[2] = "GET" }
```

Utilizzo dei modelli di filtro per associare termini nei log eventi delimitati da spazi

Scrittura di modelli di filtro per log eventi delimitati da spazi

È possibile creare modelli di filtro per verificare la presenza di termini in log eventi delimitati da spazi. Di seguito viene fornito un esempio di log eventi delimitato da spazi e viene illustrato come scrivere la sintassi per i modelli di filtro che verificano la presenza di termini in un log eventi delimitato da spazi.

> **Note**
>
> È possibile utilizzare qualsiasi espressione regolare condizionale durante la creazione di modelli di filtro che verifichino la presenza di termini nei log eventi delimitati da spazi. Per un elenco delle espressioni regolari supportate, consulta la sezione Espressioni regolari supportate.

Example: Space-delimited log event

Il seguente frammento di codice mostra un log eventi delimitato da spazi che contiene sette campi: `ip`, `user`, `username`, `timestamp`, `request`, `status_code` e `bytes`.

```
127.0.0.1 Prod frank [10/Oct/2000:13:25:15 -0700] "GET /index.html HTTP/1.0" 404
 1534
```

> **Note**
>
> I caratteri tra parentesi quadre ("[]") e virgolette doppie ("") sono considerati campi singoli.

Writing filter patterns that match terms in a space-delimited log event

Per creare un modello di filtro che verifichi la presenza di termini in un log eventi delimitato da spazi, racchiudi il modello di filtro tra parentesi quadre ("[]") e specifica i campi con nomi separati da virgole (","). Il seguente modello di filtro analizza sette campi.

```
[ip=%127\.0\.0\.[1-9]%, user, username, timestamp, request =*.html*, status_code =
 4*, bytes]
```

È possibile utilizzare operazioni numeriche (>, <, =, !=, >= oppure <=) e l'asterisco (*) come carattere jolly per fornire le condizioni del modello di filtro. Nel modello di filtro di esempio, `ip` utilizza espressioni regolari che restituiscono l'intervallo di indirizzi IP 127.0.0.1 - 127.0.0.9, `request` contiene un carattere jolly che indica che deve estrarre un valore con `.html` e `status_code` contiene un carattere jolly che indica che deve estrarre un valore che inizia con `4`.

Se non conosci il numero di campi che stai analizzando in un log eventi delimitato da spazi, puoi usare i puntini di sospensione (...) per fare riferimento a qualsiasi campo senza nome. Questi possono fare riferimento a tutti i campi necessari. L'esempio seguente mostra un modello di filtro con puntini di sospensione che rappresentano i primi quattro campi senza nome mostrati nel modello di filtro di esempio precedente.

```
[..., request =*.html*, status_code = 4*, bytes]
```

Puoi anche utilizzare gli operatori logici AND (&&) e OR (||) per creare espressioni composte. Il seguente modello di filtro contiene un'espressione composta che indica che il valore di `status_code` deve essere `404` o `410`.

```
[ip, user, username, timestamp, request =*.html*, status_code = 404 || status_code =
 410, bytes]
```

Corrispondenza di termini in log eventi delimitati da spazi utilizzando la corrispondenza dei modelli

È possibile utilizzare la corrispondenza di modelli per creare modelli di filtro delimitati da spazi che associano termini in un ordine specifico. Specifica l'ordine dei termini con gli indicatori. Utilizza w1 per rappresentare il primo termine e w2 e così via per rappresentare l'ordine dei termini successivi.

Inserisci una virgola (",") tra i termini. Gli esempi seguenti contengono frammenti di codice che mostrano come è possibile utilizzare la corrispondenza di modelli con modelli di filtro delimitati da spazi.

> **Note**
>
> È possibile utilizzare qualsiasi espressione regolare condizionale durante la creazione di modelli di filtro che verifichino la presenza di termini nei log eventi delimitati da spazi. Per un elenco delle espressioni regolari supportate, consulta la sezione Espressioni regolari supportate.

Log eventi delimitato da spazi

```
INFO 09/25/2014 12:00:00 GET /service/resource/67 1200
INFO 09/25/2014 12:00:01 POST /service/resource/67/part/111 1310
WARNING 09/25/2014 12:00:02 Invalid user request
ERROR 09/25/2014 12:00:02 Failed to process request
```

Example: Match terms in order

Il seguente modello di filtro delimitato da spazi restituisce log eventi in cui la prima parola è ERROR.

```
[w1=ERROR, w2]
```

> **Note**
>
> Quando crei modelli di filtro delimitati da spazi che utilizzano la corrispondenza di modelli, devi includere un indicatore vuoto dopo aver specificato l'ordine dei termini. Ad esempio, per creare un modello di filtro che restituisca i log eventi in cui la prima parola è ERROR, includi un indicatore vuoto w2 dopo il termine w1.

Example: Match terms with AND (&&) and OR (||)

È possibile utilizzare gli operatori logici AND ("&&") e OR ("||") per creare modelli di filtro delimitati da spazi che contengono condizioni. Il seguente modello di filtro restituisce log eventi in cui la prima parola è ERROR o WARNING.

```
[w1=ERROR || w1=WARNING, w2]
```

Example: Exclude terms from matches

È possibile creare modelli di filtro delimitati da spazi che restituiscono log eventi escludendo uno o più termini. Inserisci un simbolo non uguale ("!=") prima del termine o dei termini che desideri escludere. Il seguente frammento di codice mostra un esempio di un modello di filtro che restituisce log eventi in cui le prime parole non sono ERROR e WARNING.

```
[w1!=ERROR && w1!=WARNING, w2]
```

Example: Match the top level item in a resource URI

Il seguente frammento di codice mostra un esempio di un modello di filtro che restituisce l'elemento di livello superiore in un URI di risorsa che utilizza regex.

```
[logLevel, date, time, method, url=%/service/resource/[0-9]+$%, response_time]
```

Example: Match the child level item in a resource URI

Il seguente frammento di codice mostra un esempio di un modello di filtro che corrisponde all'elemento di livello inferiore in un URI di risorsa che utilizza regex.

```
[logLevel, date, time, method, url=%/service/resource/[0-9]+/part/[0-9]+$%,
 response_time]
```

Abilitazione della registrazione dai servizi AWS

Sebbene molti servizi pubblichino i log solo nei CloudWatch log, alcuni AWS servizi possono pubblicare i log direttamente su Amazon Simple Storage Service o Amazon Kinesis Data Firehose. Se il requisito principale per i log è lo storage o l'elaborazione in uno di questi servizi, puoi fare facilmente in modo che il servizio che produce i log li invii direttamente ad Amazon S3 o Kinesis Data Firehose senza ulteriori impostazioni.

Anche quando i log vengono pubblicati direttamente in Amazon S3 o Kinesis Data Firehose, vengono applicati costi. Per ulteriori informazioni, consulta Vending Logs nella scheda Logs di Amazon Pricing. CloudWatch

Alcuni AWS servizi utilizzano un'infrastruttura comune per inviare i propri log. Per abilitare la registrazione da questi servizi, è necessario accedere come utente con determinate autorizzazioni. Inoltre, è necessario concedere le autorizzazioni AWS per consentire l'invio dei log.

Per i servizi che richiedono queste autorizzazioni, sono necessarie due versioni delle autorizzazioni. I servizi che richiedono queste autorizzazioni aggiuntive sono indicati come [Autorizzazioni V1] supportate e [Autorizzazioni V2] supportate nella tabella. Per informazioni su queste autorizzazioni richieste, consulta le sezioni dopo la tabella.

Tipo di log	CloudWatch Logs	Amazon S3	Kinesis Data Firehose
Log di accesso Gateway Amazon API	[Autorizzazioni V1] supportate		
AWS AppSync logs	Supportato		
Log di Amazon Aurora MySQL	Supportato		
Log di parametri di qualità dei supporti Amazon Chime e log dei messaggi SIP	[Autorizzazioni V1] supportate		

Tipo di log	CloudWatch Logs	Amazon S3	Kinesis Data Firehose
CloudFront: registri di accesso		[Autorizzazioni V1] supportate	
AWS CloudHSM registri di controllo	Supportato		
CloudWatch Evidentemente i registri degli eventi di valutazione	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate	
CloudWatch Registri di Internet Monitor		[Autorizzazioni V1] supportate	
CloudTrail registri	Supportato		
AWS CodeBuild logs	Supportato		
Amazon Cognito logs	[Autorizzazioni V1] supportate		
Log di Amazon Connect	Supportato		
AWS DataSync logs	Supportato		
Log ElastiCache di Amazon per Redis	[Autorizzazioni V1] supportate		[Autorizzazioni V1] supportate
AWS Elastic Beanstalk logs	Supportato		
Log di Amazon Elastic Container Service	Supportato		
Log del piano di controllo (control-plane) Amazon Elastic Kubernetes Service	Supportato		

Tipo di log	CloudWatch Logs	Amazon S3	Kinesis Data Firehose
AWS Fargate logs	Supportato		
AWS Fault Injection Service registri degli esperimenti		[Autorizzazioni V1] supportate	
Amazon FinSpace	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate
AWS Global Accelerator registri di flusso		[Autorizzazioni V1] supportate	
AWS Glue registri di lavoro	Supportato		
Log delle chat di Amazon Interactive Video Service Chat	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate
AWS IoT logs	Supportato		
AWS IoT FleetWise logs	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate
AWS Lambda logs	Supportato		
Log di Amazon Macie	Supportato		
AWS Mainframe Modernization	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate

Tipo di log	CloudWatch Logs	Amazon S3	Kinesis Data Firehose
Log di Amazon Managed Service for Prometheus	[Autorizzazioni V1] supportate		
Log di broker Amazon MSK	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate
Log di Amazon MSK Connect	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate
Log generali e di controllo di Amazon MQ	Supportato		
AWS Registri del Network Firewall	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate
Log di accesso di Network Load Balancer		[Autorizzazioni V1] supportate	
OpenSearch registri	Supportato		
Registri OpenSearch di ingestione di Amazon Service	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate
AWS OpsWorks logs	Supportato		
Log ServicePostgre SQL di Amazon Relational Database	Supportato		
AWS RoboMaker registri	Supportato		
Log di query DNS pubblici di Amazon Route 53	Supportato		

Tipo di log	CloudWatch Logs	Amazon S3	Kinesis Data Firehose
Log delle query del risolutore Amazon Route 53	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate	
SageMaker Eventi Amazon	[Autorizzazioni V1] supportate		
Eventi per i SageMaker lavoratori di Amazon	[Autorizzazioni V1] supportate		
AWS Registri VPN da sito a sito	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate
Log di Amazon Simple Notification Service	Supportato		
Log delle policy di protezione dei dati di Amazon Simple Notification Service	Supportato		
File di feed di dati dell'istanza Spot EC2		[Autorizzazioni V1] supportate	
AWS Step Functions Registri Express Workflow e Standard Workflow	[Autorizzazioni V1] supportate		
Log di controllo e log di integrità del Gateway di archiviazione	[Autorizzazioni V1] supportate		
AWS Transfer Family logs	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate

Tipo di log	CloudWatch Logs	Amazon S3	Kinesis Data Firehose
Accesso verificato da AWS logs	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate
Flussi di log Amazon Virtual Private Cloud		[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate
Log di accesso Amazon VPC Lattice	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate
AWS WAF logs	[Autorizzazioni V1] supportate	[Autorizzazioni V1] supportate	Supportato
Amazon CodeWhisperer	[Autorizzazioni V2] supportate	[Autorizzazioni V2] supportate	[Autorizzazioni V2] supportate

Registrazione che richiede autorizzazioni aggiuntive [V1]

Alcuni AWS servizi utilizzano un'infrastruttura comune per inviare i propri log a CloudWatch Logs, Amazon S3 o Kinesis Data Firehose. Per abilitare i servizi AWS elencati nella tabella seguente per inviare i log a queste destinazioni, devi essere connesso come un utente che dispone di determinate autorizzazioni.

Inoltre, è necessario concedere le autorizzazioni per consentire l'invio AWS dei log. AWS può creare automaticamente tali autorizzazioni al momento della configurazione dei registri oppure è possibile crearle personalmente prima di configurare la registrazione.

Se si sceglie di impostare AWS automaticamente le autorizzazioni e le politiche relative alle risorse necessarie quando l'utente o un membro dell'organizzazione configura per la prima volta l'invio dei log, l'utente che sta configurando l'invio dei log deve disporre di determinate autorizzazioni, come

spiegato più avanti in questa sezione. In alternativa, è possibile creare autonomamente le policy delle risorse e quindi gli utenti che impostano l'invio dei log non necessitano di altrettante autorizzazioni.

Nella tabella seguente vengono riepilogati i tipi di log e le destinazioni di log a cui si applicano le informazioni contenute in questa sezione.

Nelle sezioni seguenti vengono fornite ulteriori dettagli per ciascuna di queste destinazioni.

Registri inviati a Logs CloudWatch

> **Important**
>
> Quando si impostano i tipi di registro nell'elenco seguente per l'invio ai CloudWatch registri, AWS crea o modifica le politiche delle risorse associate al gruppo di log che riceve i registri, se necessario. Continua a leggere questa sezione per vedere i dettagli.

Questa sezione si applica quando i tipi di log elencati nella tabella della sezione precedente vengono inviati a Logs: CloudWatch

Autorizzazioni degli utenti

Per poter configurare l'invio di uno di questi tipi di log a CloudWatch Logs per la prima volta, è necessario accedere a un account con le seguenti autorizzazioni.

- `logs:CreateLogDelivery`
- `logs:PutResourcePolicy`
- `logs:DescribeResourcePolicies`
- `logs:DescribeLogGroups`

Se uno di questi tipi di log viene già inviato a un gruppo di log in CloudWatch Logs, per configurare l'invio di un altro di questi tipi di log allo stesso gruppo di log è sufficiente l'autorizzazione. `logs:CreateLogDelivery`

Policy delle risorse del gruppo di log

Il gruppo di log in cui vengono inviati i log deve disporre di una policy delle risorse che includa determinate autorizzazioni. Se il gruppo di log attualmente non dispone di un criterio

in materia di risorse e l'utente che configura la `logs:PutResourcePolicy` registrazione dispone dei `logs:DescribeLogGroups` permessi e per il gruppo di log, crea AWS automaticamente la seguente politica quando si inizia a inviare i log a CloudWatch Logs. `logs:DescribeResourcePolicies`

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AWSLogDeliveryWrite20150319",
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "delivery.logs.amazonaws.com"
        ]
      },
      "Action": [
        "logs:CreateLogStream",
        "logs:PutLogEvents"
      ],
      "Resource": [
        "arn:aws:logs:us-east-1:0123456789:log-group:my-log-group:log-stream:*"
      ],
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": ["0123456789"]
        },
        "ArnLike": {
          "aws:SourceArn": ["arn:aws:logs:us-east-1:0123456789:*"]
        }
      }
    }
  ]
}
```

Se il gruppo di log dispone di una policy delle risorse, ma tale policy non contiene l'istruzione mostrata nel policy precedente e l'utente che imposta la registrazione ha le autorizzazioni `logs:PutResourcePolicy`, `logs:DescribeResourcePolicies`, e `logs:DescribeLogGroups` per il gruppo di log, tale istruzione viene aggiunta al policy delle risorse del gruppo di log.

Considerazioni relative al limite delle dimensioni delle policy delle risorse del gruppo di log

Questi servizi devono elencare ogni gruppo di log a cui inviano i log nella politica delle risorse e le politiche delle risorse di CloudWatch Logs sono limitate a 5120 caratteri. Un servizio che invia i log a un numero elevato di gruppi di log potrebbe rientrare in questo limite.

Per mitigare questo problema, CloudWatch Logs monitora la dimensione delle politiche relative alle risorse utilizzate dal servizio che invia i log e, quando rileva che una policy si avvicina al limite di dimensione di 5120 caratteri, CloudWatch Logs abilita automaticamente la politica delle risorse per quel servizio. `/aws/vendedlogs/*` Quindi puoi iniziare a utilizzare gruppi di log con nomi che iniziano con `/aws/vendedlogs/` come destinazioni per i log di questi servizi.

Log inviati ad Amazon S3

> **Important**
>
> Quando configuri i tipi di log nel seguente elenco per l'invio ad Amazon S3, AWS crea o modifica le politiche delle risorse associate al bucket S3 che riceve i log, se necessario. Continua a leggere questa sezione per vedere i dettagli.

Questa sezione si applica quando i seguenti tipi di log vengono inviati ad Amazon S3:

- CloudFront log di accesso e log di accesso in streaming. CloudFront utilizza un modello di autorizzazioni diverso rispetto agli altri servizi in questo elenco. Per ulteriori informazioni, consulta Autorizzazioni richieste per configurare la registrazione standard e per accedere ai file di log.
- Feed di dati delle istanze Spot Amazon EC2
- AWS Global Accelerator registri di flusso
- Log del broker per Amazon Managed Streaming for Apache Kafka
- Log di accesso di Network Load Balancer
- AWS Registri del Network Firewall
- Flussi di log Amazon Virtual Private Cloud

I log pubblicati in Amazon S3 vengono pubblicati in un bucket esistente da te specificato. Uno o più file di log vengono creati ogni cinque minuti nel bucket specificato.

Quando invii i log per la prima volta a un bucket Amazon S3, il servizio che consegna i log registra il proprietario del bucket per assicurarsi che i log vengano consegnati solo a un bucket appartenente a

questo account. Di conseguenza, per modificare il proprietario del bucket Amazon S3, devi ricreare o aggiornare la sottoscrizione del log nel servizio di origine.

Autorizzazioni degli utenti

Per poter configurare l'invio di uno di questi tipi di log ad Amazon S3 per la prima volta, devi aver effettuato l'accesso a un account con le seguenti autorizzazioni.

- `logs:CreateLogDelivery`
- `S3:GetBucketPolicy`
- `S3:PutBucketPolicy`

Se uno qualsiasi di questi tipi di log è già stato inviato a un bucket Amazon S3, quindi per impostare l'invio di un altro di questi tipi di log allo stesso bucket è sufficiente disporre del autorizzazione `logs:CreateLogDelivery`.

Policy delle risorse di bucket S3

Il bucket S3 in cui vengono inviati i log deve disporre di una policy delle risorse che include determinate autorizzazioni. Se il bucket attualmente non dispone di una politica delle risorse e l'utente che configura la registrazione dispone delle autorizzazioni `S3:GetBucketPolicy` e `S3:PutBucketPolicy` delle autorizzazioni per il bucket, crea AWS automaticamente la seguente politica quando inizi a inviare i log ad Amazon S3.

```
{
    "Version": "2012-10-17",
    "Id": "AWSLogDeliveryWrite20150319",
    "Statement": [
        {
            "Sid": "AWSLogDeliveryAclCheck",
            "Effect": "Allow",
            "Principal": {
                "Service": "delivery.logs.amazonaws.com"
                },
            "Action": "s3:GetBucketAcl",
            "Resource": "arn:aws:s3:::my-bucket",
            "Condition": {
                "StringEquals": {
                "aws:SourceAccount": ["0123456789"]
                },
```

```
                "ArnLike": {
                "aws:SourceArn": ["arn:aws:logs:us-east-1:0123456789:*"]
                }
            }
        },
        {
            "Sid": "AWSLogDeliveryWrite",
            "Effect": "Allow",
            "Principal": {
                "Service": "delivery.logs.amazonaws.com"
            },
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::my-bucket/AWSLogs/account-ID/*",
            "Condition": {
                "StringEquals": {
                    "s3:x-amz-acl": "bucket-owner-full-control",
                    "aws:SourceAccount": ["0123456789"]
                },
                "ArnLike": {
                    "aws:SourceArn": ["arn:aws:logs:us-east-1:0123456789:*"]
                }
            }
        }
    ]
}
```

Nella policy precedente, per `aws:SourceAccount`, specifica l'elenco degli ID account per i quali i log vengono consegnati a questo bucket. Per `aws:SourceArn`, specifica l'elenco di ARN della risorsa che genera i log, nel formato `arn:aws:logs:`*`source-region`*`:`*`source-account-id`*`:*`.

Se il bucket dispone di una policy delle risorse, ma tale criterio non contiene l'istruzione mostrata nella policy precedente e l'utente che imposta la registrazione ha le autorizzazioni `S3:GetBucketPolicy` e `S3:PutBucketPolicy` per il bucket, tale istruzione viene aggiunta alla policy delle risorse del bucket.

> **Note**
>
> In alcuni casi, potresti riscontrare `AccessDenied` degli errori AWS CloudTrail se l'`s3:ListBucket`autorizzazione non è stata concessa. `delivery.logs.amazonaws.com` Per evitare questi errori nei CloudTrail log, è necessario concedere l'`s3:ListBucket`autorizzazione `delivery.logs.amazonaws.com` e includere `Condition` i parametri mostrati con l'`s3:GetBucketAcl`autorizzazione impostata nella

precedente policy del bucket. Per renderlo più semplice, invece di creare una nuova `Statement`, puoi aggiornar direttamente`AWSLogDeliveryAclCheck` per essere `“Action”: [“s3:GetBucketAcl”, “s3:ListBucket”]`

Crittografia lato server di bucket Amazon S3

Puoi proteggere i dati nel tuo bucket Amazon S3 abilitando la crittografia lato server con chiavi gestite da Amazon S3 (SSE-S3) o la crittografia lato server con una chiave archiviata in (SSE-KMS). AWS KMS AWS Key Management Service Per ulteriori informazioni, consulta Protezione dei dati con la crittografia lato server.

Se si sceglie SSE-S3, non è richiesta alcuna configurazione aggiuntiva. Amazon S3 gestisce la chiave di crittografia.

Warning

Se scegli SSE-KMS, devi utilizzare una chiave gestita dal cliente, poiché l'utilizzo di una chiave gestita non è supportato in questo scenario. AWS Se si configura la crittografia utilizzando una chiave AWS gestita, i log verranno consegnati in un formato illeggibile.

Quando utilizzi una AWS KMS chiave gestita dal cliente, puoi specificare l'Amazon Resource Name (ARN) della chiave gestita dal cliente quando abiliti la crittografia dei bucket. È necessario aggiungere quanto segue alla policy della chiave per la chiave gestita dal cliente (non alla policy di bucket per il bucket S3), in modo che l'account di consegna del log possa scrivere nel bucket S3.

Se scegli SSE-KMS, devi utilizzare una chiave gestita dal cliente, poiché l'utilizzo di una chiave AWS gestita non è supportato in questo scenario. Quando utilizzi una AWS KMS chiave gestita dal cliente, puoi specificare l'Amazon Resource Name (ARN) della chiave gestita dal cliente quando abiliti la crittografia dei bucket. È necessario aggiungere quanto segue alla policy della chiave per la chiave gestita dal cliente (non alla policy di bucket per il bucket S3), in modo che l'account di consegna del log possa scrivere nel bucket S3.

```
{
    "Sid": "Allow Logs Delivery to use the key",
    "Effect": "Allow",
    "Principal": {
        "Service": [ "delivery.logs.amazonaws.com" ]
```

```
    },
    "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "aws:SourceAccount": ["0123456789"]
        },
        "ArnLike": {
            "aws:SourceArn": ["arn:aws:logs:us-east-1:0123456789:*"]
        }
        }
}
```

Per `aws:SourceAccount`, specifica l'elenco degli ID account per i quali i log vengono consegnati a questo bucket. Per `aws:SourceArn`, specifica l'elenco di ARN della risorsa che genera i registri, nel formato `arn:aws:logs:`*`source-region`*`:`*`source-account-id`*`:*`.

Log inviati a Kinesis Data Firehose

Questa sezione si applica quando i tipi di log elencati nella tabella della sezione precedente vengono inviati a Kinesis Data Firehose:

Autorizzazioni degli utenti

Per poter configurare l'invio di uno di questi tipi di log ad Kinesis Data Firehose per la prima volta, devi aver effettuato l'accesso a un account con le seguenti autorizzazioni.

- `logs:CreateLogDelivery`
- `firehose:TagDeliveryStream`
- `iam:CreateServiceLinkedRole`

Se uno qualsiasi di questi tipi di log è già stato inviato a Kinesis Data Firehose, quindi per impostare l'invio di un altro di questi tipi di log a Kinesis Data Firehose è necessario disporre solo delle autorizzazioni `logs:CreateLogDelivery` e `firehose:TagDeliveryStream`.

Ruoli IAM utilizzati per le autorizzazioni

Poiché Kinesis Data Firehose non utilizza AWS policy relative alle risorse, utilizza i ruoli IAM per configurare questi log da inviare a Kinesis Data Firehose. AWS crea un ruolo collegato al servizio denominato. AWSServiceRoleForLogDelivery Questo ruolo collegato al servizio include le autorizzazioni seguenti.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "firehose:PutRecord",
                "firehose:PutRecordBatch",
                "firehose:ListTagsForDeliveryStream"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/LogDeliveryEnabled": "true"
                }
            },
            "Effect": "Allow"
        }
    ]
}
```

Questo ruolo collegato al servizio concede l'autorizzazione per tutti i flussi di distribuzione di Kinesis Data Firehose con il tag impostato su. `LogDeliveryEnabled` `true` AWS assegna questo tag al flusso di consegna di destinazione quando configuri la registrazione.

Questo ruolo collegato al servizio dispone inoltre di una policy di attendibilità che autorizzi il principale del servizio `delivery.logs.amazonaws.com` di assumere il ruolo collegato al servizio necessario. Questa policy di attendibilità è la seguente:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "delivery.logs.amazonaws.com"
```

```
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Registrazione che richiede autorizzazioni aggiuntive [V2]

Alcuni AWS servizi utilizzano un nuovo metodo per inviare i log. Si tratta di un metodo flessibile che consente di configurare la consegna dei log da questi servizi verso una o più delle seguenti destinazioni: CloudWatch Logs, Amazon S3 o Kinesis Data Firehose.

Per configurare la consegna dei log tra un AWS servizio supportato e una destinazione, devi fare quanto segue:

- Crea un'origine di consegna, ossia un oggetto logico che rappresenta la risorsa che sta effettivamente inviando i log. Per ulteriori informazioni, vedere PutDeliverySource.
- Crea una destinazione di consegna, ossia un oggetto logico che rappresenta la destinazione di consegna effettiva. Per ulteriori informazioni, vedere PutDeliveryDestination.
- Se si inviano registri su più account, è necessario utilizzarli PutDeliveryDestinationPolicynell'account di destinazione per assegnare una IAM politica alla destinazione. Questa policy consente la consegna a quella destinazione.
- Utilizza `CreateDelivery` per creare una consegna associando esattamente un'origine di consegna e una destinazione di consegna.

Le sezioni seguenti forniscono i dettagli delle autorizzazioni necessarie quando si effettua l'accesso per configurare la consegna dei log a ciascun tipo di destinazione utilizzando il processo V2. Queste autorizzazioni possono essere concesse a un ruolo IAM con cui hai effettuato l'accesso.

Oltre alle autorizzazioni elencate nelle seguenti sezioni, se stai configurando la consegna dei log utilizzando la console anziché le API, hai bisogno anche delle seguenti autorizzazioni aggiuntive:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowLogDeliveryActions",
            "Effect": "Allow",
            "Action": [
```

```
                "logs:DescribeLogGroups",
                "firehose:ListDeliveryStreams",
                "firehose:DescribeDeliveryStream",
                "s3:ListAllMyBuckets",
                "s3:ListBucket",
                "s3:GetBucketLocation"
            ],
            "Resource": [
                "arn:aws:logs:region:account-id:log-group:*",
                "arn:aws:firehose:region:account-id:deliverystream/*",
                "arn:aws:s3:::*"
            ]
        }
    ]
}
```

Registri inviati a Logs CloudWatch

Autorizzazioni degli utenti

Per abilitare l'invio dei log ai CloudWatch registri, è necessario accedere con le seguenti autorizzazioni.

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Sid": "AllowLogDeliveryActions",
        "Effect": "Allow",
        "Action": [
            "logs:PutDeliverySource",
            "logs:GetDeliverySource",
            "logs:DeleteDeliverySource",
            "logs:DescribeDeliverySources",
            "logs:PutDeliveryDestination",
            "logs:GetDeliveryDestination",
            "logs:DeleteDeliveryDestination",
            "logs:DescribeDeliveryDestinations",
            "logs:CreateDelivery",
            "logs:GetDelivery",
            "logs:DeleteDelivery",
            "logs:DescribeDeliveries",
            "logs:PutDeliveryDestinationPolicy",
            "logs:GetDeliveryDestinationPolicy",
```

```
            "logs:DeleteDeliveryDestinationPolicy"
        ],
        "Resource": [
            "arn:aws:logs:region:account-id:delivery-source:*",
            "arn:aws:logs:region:account-id:delivery:*",
            "arn:aws:logs:region:account-id:delivery-destination:*"
        ]
    },
    {
        "Sid": "AllowUpdatesToResourcePolicyCWL",
        "Effect": "Allow",
        "Action": [
            "logs:PutResourcePolicy",
            "logs:DescribeResourcePolicies",
            "logs:DescribeLogGroups"
        ],
        "Resource": [
            "arn:aws:logs:region:account-id:log-group:*"
        ]
    }]
}
```

Policy delle risorse del gruppo di log

Il gruppo di log in cui vengono inviati i log deve disporre di una policy delle risorse che includa determinate autorizzazioni. Se al momento il gruppo di log non dispone di una politica in materia di risorse e l'utente che configura la `logs:PutResourcePolicy` registrazione dispone `logs:DescribeLogGroups` delle autorizzazioni relative al gruppo di log, crea AWS automaticamente la seguente politica quando si inizia a inviare i log a Logs. `logs:DescribeResourcePolicies` CloudWatch

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AWSLogDeliveryWrite20150319",
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "delivery.logs.amazonaws.com"
        ]
      },
      "Action": [
```

```
        "logs:CreateLogStream",
        "logs:PutLogEvents"
      ],
      "Resource": [
        "arn:aws:logs:us-east-1:0123456789:log-group:my-log-group:log-stream:*"
      ],
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": ["0123456789"]
        },
        "ArnLike": {
          "aws:SourceArn": ["arn:aws:logs:us-east-1:0123456789:*"]
        }
      }
    }
  ]
}
```

Considerazioni relative al limite delle dimensioni delle policy delle risorse del gruppo di log

Questi servizi devono elencare ogni gruppo di log a cui inviano i log nella politica delle risorse e le politiche delle risorse di CloudWatch Logs sono limitate a 5120 caratteri. Un servizio che invia i log a un numero elevato di gruppi di log potrebbe rientrare in questo limite.

Per mitigare questo problema, CloudWatch Logs monitora la dimensione delle politiche relative alle risorse utilizzate dal servizio che invia i log e, quando rileva che una policy si avvicina al limite di dimensione di 5120 caratteri, CloudWatch Logs abilita automaticamente la politica delle risorse per quel servizio. `/aws/vendedlogs/*` Quindi puoi iniziare a utilizzare gruppi di log con nomi che iniziano con `/aws/vendedlogs/` come destinazioni per i log di questi servizi.

Log inviati ad Amazon S3

Autorizzazioni degli utenti

Per abilitare l'invio di log ad Amazon S3, devi aver effettuato l'accesso con le seguenti autorizzazioni.

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Sid": "AllowLogDeliveryActions",
        "Effect": "Allow",
        "Action": [
            "logs:PutDeliverySource",
```

```
            "logs:GetDeliverySource",
            "logs:DeleteDeliverySource",
            "logs:DescribeDeliverySources",
            "logs:PutDeliveryDestination",
            "logs:GetDeliveryDestination",
            "logs:DeleteDeliveryDestination",
            "logs:DescribeDeliveryDestinations",
            "logs:CreateDelivery",
            "logs:GetDelivery",
            "logs:DeleteDelivery",
            "logs:DescribeDeliveries",
            "logs:PutDeliveryDestinationPolicy",
            "logs:GetDeliveryDestinationPolicy",
            "logs:DeleteDeliveryDestinationPolicy"
        ],
        "Resource": [
            "arn:aws:logs:region:account-id:delivery-source:*",
            "arn:aws:logs:region:account-id:delivery:*",
            "arn:aws:logs:region:account-id:delivery-destination:*"
        ]
    },
    {
            "Sid": "AllowUpdatesToResourcePolicyS3",
            "Effect": "Allow",
            "Action": [
                "s3:PutBucketPolicy",
                "s3:GetBucketPolicy"
            ],
            "Resource": [
                "arn:aws:s3:::bucket_name"
            ]
        }]
}
```

Il bucket S3 in cui vengono inviati i log deve disporre di una policy delle risorse che include determinate autorizzazioni. Se il bucket attualmente non dispone di una politica delle risorse e l'utente che configura la registrazione dispone delle autorizzazioni `S3:GetBucketPolicy` e `S3:PutBucketPolicy` delle autorizzazioni per il bucket, crea AWS automaticamente la seguente politica quando inizi a inviare i log ad Amazon S3.

```
{
    "Version": "2012-10-17",
    "Id": "AWSLogDeliveryWrite20150319",
```

```
    "Statement": [
        {
            "Sid": "AWSLogDeliveryAclCheck",
            "Effect": "Allow",
            "Principal": {
                "Service": "delivery.logs.amazonaws.com"
                },
            "Action": "s3:GetBucketAcl",
            "Resource": "arn:aws:s3:::my-bucket",
            "Condition": {
                "StringEquals": {
                "aws:SourceAccount": ["0123456789"]
                },
                "ArnLike": {
                "aws:SourceArn": ["arn:aws:logs:us-east-1:0123456789:delivery-source*"]
                }
            }
        },
        {
            "Sid": "AWSLogDeliveryWrite",
            "Effect": "Allow",
            "Principal": {
                "Service": "delivery.logs.amazonaws.com"
            },
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::my-bucket/AWSLogs/account-ID/*",
            "Condition": {
                "StringEquals": {
                    "s3:x-amz-acl": "bucket-owner-full-control",
                    "aws:SourceAccount": ["0123456789"]
                },
                "ArnLike": {
                    "aws:SourceArn": ["arn:aws:logs:us-east-1:0123456789:delivery-
source:*"]
                }
            }
        }
    ]
}
```

Nella policy precedente, per `aws:SourceAccount`, specifica l'elenco degli ID account per i quali i log vengono consegnati a questo bucket. Per `aws:SourceArn`, specifica l'elenco di ARN della risorsa che genera i log, nel formato `arn:aws:logs:`*`source-region`*`:`*`source-account-id`*`:*`.

Se il bucket dispone di una policy delle risorse, ma tale criterio non contiene l'istruzione mostrata nella policy precedente e l'utente che imposta la registrazione ha le autorizzazioni `S3:GetBucketPolicy` e `S3:PutBucketPolicy` per il bucket, tale istruzione viene aggiunta alla policy delle risorse del bucket.

> **Note**
>
> In alcuni casi, potresti riscontrare `AccessDenied` degli errori AWS CloudTrail se l'`s3:ListBucket`autorizzazione non è stata concessa. `delivery.logs.amazonaws.com` Per evitare questi errori nei CloudTrail log, è necessario concedere l'`s3:ListBucket`autorizzazione `delivery.logs.amazonaws.com` e includere `Condition` i parametri mostrati con l'`s3:GetBucketAcl`autorizzazione impostata nella precedente policy del bucket. Per renderlo più semplice, invece di creare una nuova `Statement`, puoi aggiornar direttamente`AWSLogDeliveryAclCheck` per essere `“Action”: [“s3:GetBucketAcl”, “s3:ListBucket”]`

Crittografia lato server di bucket Amazon S3

Puoi proteggere i dati nel tuo bucket Amazon S3 abilitando la crittografia lato server con chiavi gestite da Amazon S3 (SSE-S3) o la crittografia lato server con una chiave archiviata in (SSE-KMS). AWS KMS AWS Key Management Service Per ulteriori informazioni, consulta Protezione dei dati con la crittografia lato server.

Se si sceglie SSE-S3, non è richiesta alcuna configurazione aggiuntiva. Amazon S3 gestisce la chiave di crittografia.

> **Warning**
>
> Se scegli SSE-KMS, devi utilizzare una chiave gestita dal cliente, poiché l'utilizzo di una chiave gestita non è supportato in questo scenario. AWS Se si configura la crittografia utilizzando una chiave AWS gestita, i log verranno consegnati in un formato illeggibile.

Quando utilizzi una AWS KMS chiave gestita dal cliente, puoi specificare l'Amazon Resource Name (ARN) della chiave gestita dal cliente quando abiliti la crittografia dei bucket. È necessario aggiungere quanto segue alla policy della chiave per la chiave gestita dal cliente (non alla policy di bucket per il bucket S3), in modo che l'account di consegna del log possa scrivere nel bucket S3.

Se scegli SSE-KMS, devi utilizzare una chiave gestita dal cliente, poiché l'utilizzo di una chiave AWS gestita non è supportato in questo scenario. Quando utilizzi una AWS KMS chiave gestita dal cliente, puoi specificare l'Amazon Resource Name (ARN) della chiave gestita dal cliente quando abiliti la crittografia dei bucket. È necessario aggiungere quanto segue alla policy della chiave per la chiave gestita dal cliente (non alla policy di bucket per il bucket S3), in modo che l'account di consegna del log possa scrivere nel bucket S3.

```
{
    "Sid": "Allow Logs Delivery to use the key",
    "Effect": "Allow",
    "Principal": {
        "Service": [ "delivery.logs.amazonaws.com" ]
    },
    "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "aws:SourceAccount": ["0123456789"]
        },
        "ArnLike": {
            "aws:SourceArn": ["arn:aws:logs:us-east-1:0123456789:delivery-source:*"]
        }
        }
}
```

Per `aws:SourceAccount`, specifica l'elenco degli ID account per i quali i log vengono consegnati a questo bucket. Per `aws:SourceArn`, specifica l'elenco di ARN della risorsa che genera i registri, nel formato `arn:aws:logs:`*`source-region`*`:`*`source-account-id`*`:*`.

Log inviati a Kinesis Data Firehose

Autorizzazioni degli utenti

Per abilitare l'invio di log a Kinesis Data Firehose, devi aver effettuato l'accesso con le seguenti autorizzazioni.

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Sid": "AllowLogDeliveryActions",
        "Effect": "Allow",
        "Action": [
            "logs:PutDeliverySource",
            "logs:GetDeliverySource",
            "logs:DeleteDeliverySource",
            "logs:DescribeDeliverySources",
            "logs:PutDeliveryDestination",
            "logs:GetDeliveryDestination",
            "logs:DeleteDeliveryDestination",
            "logs:DescribeDeliveryDestinations",
            "logs:CreateDelivery",
            "logs:GetDelivery",
            "logs:DeleteDelivery",
            "logs:DescribeDeliveries",
            "logs:PutDeliveryDestinationPolicy",
            "logs:GetDeliveryDestinationPolicy",
            "logs:DeleteDeliveryDestinationPolicy"
        ],
        "Resource": [
            "arn:aws:logs:region:account-id:delivery-source:*",
            "arn:aws:logs:region:account-id:delivery:*",
            "arn:aws:logs:region:account-id:delivery-destination:*"
        ]
    },
    {
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowUpdatesToResourcePolicyFH",
            "Effect": "Allow",
            "Action": [
                "firehose:TagDeliveryStream",
                "iam:CreateServiceLinkedRole"
            ],
            "Resource": [
                "arn:aws:firehose:region:account-id:deliverystream/delivery-stream-
name"
            ]
        }
```

```
    ]
}]
}
```

Ruoli IAM utilizzati per le autorizzazioni delle risorse

Poiché Kinesis Data Firehose non utilizza AWS policy relative alle risorse, utilizza i ruoli IAM per configurare questi log da inviare a Kinesis Data Firehose. AWS crea un ruolo collegato al servizio denominato. AWSServiceRoleForLogDelivery Questo ruolo collegato al servizio include le autorizzazioni seguenti.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "firehose:PutRecord",
                "firehose:PutRecordBatch",
                "firehose:ListTagsForDeliveryStream"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/LogDeliveryEnabled": "true"
                }
            },
            "Effect": "Allow"
        }
    ]
}
```

Questo ruolo collegato al servizio concede l'autorizzazione per tutti i flussi di distribuzione di Kinesis Data Firehose con il tag impostato su. `LogDeliveryEnabled` `true` AWS assegna questo tag al flusso di consegna di destinazione quando configuri la registrazione.

Questo ruolo collegato al servizio dispone inoltre di una policy di attendibilità che autorizzi il principale del servizio `delivery.logs.amazonaws.com` di assumere il ruolo collegato al servizio necessario. Questa policy di attendibilità è la seguente:

```
{
  "Version": "2012-10-17",
```

```
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "delivery.logs.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Prevenzione del confused deputy tra servizi

Con "confused deputy" si intende un problema di sicurezza in cui un'entità che non dispone dell'autorizzazione per eseguire una certa operazione può costringere un'entità con più privilegi a eseguire tale operazione. Nel frattempo AWS, l'impersonificazione tra servizi può portare al confuso problema del vicesceriffo. La rappresentazione tra servizi può verificarsi quando un servizio (il servizio chiamante) effettua una chiamata a un altro servizio (il servizio chiamato). Il servizio chiamante può essere manipolato per utilizzare le proprie autorizzazioni e agire sulle risorse di un altro cliente, a cui normalmente non avrebbe accesso. Per evitare ciò, AWS fornisce strumenti per poterti a proteggere i tuoi dati per tutti i servizi con entità di servizio a cui è stato concesso l'accesso alle risorse del tuo account.

Consigliamo di utilizzare le chiavi di contesto `aws:SourceArn`e `aws:SourceAccount`global condition nelle politiche delle risorse per limitare le autorizzazioni concesse da CloudWatch Logs e Amazon S3 ai servizi che generano i log. Se si utilizzano entrambe le chiavi di contesto delle condizioni globali, il valore `aws:SourceAccount` e l'account nel valore `aws:SourceArn` devono utilizzare lo stesso ID account nella stessa istruzione di policy.

I valori di `aws:SourceArn` devono essere gli ARN delle origini di consegna che generano log.

Il modo più efficace per proteggersi dal problema "confused deputy" è quello di usare la chiave di contesto della condizione globale `aws:SourceArn` con l'ARN completo della risorsa. Se non conosci l'ARN completo della risorsa o scegli più risorse, utilizza la chiave di contesto della condizione globale `aws:SourceArn` con caratteri jolly (`*`) per le parti sconosciute dell'ARN.

Le policy nelle sezioni precedenti di questa pagina mostrano come è possibile utilizzare le chiavi di contesto delle condizioni globali `aws:SourceArn` e `aws:SourceAccount` per prevenire il problema "confused deputy".

CloudWatch Registra gli aggiornamenti delle politiche gestite AWS

Visualizza i dettagli sugli aggiornamenti alle politiche AWS gestite per CloudWatch Logs da quando questo servizio ha iniziato a tenere traccia di queste modifiche. Per ricevere avvisi automatici sulle modifiche a questa pagina, iscriviti al feed RSS nella pagina della cronologia dei documenti dei CloudWatch registri.

Modifica	Descrizione	Data
AWSServiceRoleForLogDelivery politica relativa ai ruoli collegati ai servizi: aggiornamento a una politica esistente	CloudWatch I log hanno modificato le autorizzazioni nella policy IAM associata al ruolo collegato al servizio. AWSServiceRoleForLogDelivery È stata apportata la seguente modifica: • La chiave di condizione `firehose:ResourceTag/LogDeliveryEnabled": "true"` è stata modificata in `aws:ResourceTag/LogDeliveryEnabled": "true"` .	15 luglio 2021
CloudWatch I log hanno iniziato a tenere traccia delle modifiche	CloudWatch Logs ha iniziato a tenere traccia delle modifiche relative alle politiche AWS gestite.	10 giugno 2021

Esportazione di dati di log in Amazon S3

Esporta dati di log dai gruppi di log in un bucket Amazon S3 e utilizzali nell'elaborazione e nell'analisi personalizzate, oppure caricali in altri sistemi. È possibile eseguire l'esportazione in un bucket nello stesso account o in un account diverso.

Puoi eseguire le operazioni indicate di seguito:

- Esporta i dati di registro in bucket S3 crittografati da SSE-KMS in () AWS Key Management Service AWS KMS
- Esportazione dei dati di log in bucket S3 per i quali S3 Object Lock è abilitato con un periodo di conservazione

Per iniziare il processo di esportazione, è necessario creare un bucket S3 per archiviare i dati di log esportati. Puoi archiviare i file esportati nel bucket S3 e definire regole del ciclo di vita Amazon S3 per archiviare o eliminare i file esportati automaticamente.

Puoi eseguire l'esportazione in bucket S3 che sono crittografati con AES-256 o con SSE-KMS. L'esportazione in bucket crittografati con DSSE-KMS non è supportata.

Puoi esportare i log da più gruppi di log o da più intervalli di tempo nello stesso bucket S3. Per separare i dati di log per ogni attività di esportazione, puoi specificare un prefisso che verrà utilizzato come prefisso della chiave Amazon S3 per tutti gli oggetti esportati.

> **Note**
>
> L'ordinamento basato sul tempo su blocchi di dati di log all'interno di un file esportato non è garantito. Puoi ordinare i dati del campo di log esportati utilizzando le utilità Linux. Ad esempio, il seguente comando di utilità ordina gli eventi in tutti i file `.gz` in una singola cartella.
>
> ```
> find . -exec zcat {} + | sed -r 's/^[0-9]+/\x0&/' | sort -z
> ```
>
> Il seguente comando di utilità ordina i file .gz da più sottocartelle.
>
> ```
> find ./*/ -type f -exec zcat {} + | sed -r 's/^[0-9]+/\x0&/' | sort -z
> ```

Inoltre, puoi usare un altro comando `stdout` per reindirizzare l'output ordinato a un altro file per salvarlo.

Potrebbero essere necessarie fino a 12 ore affinché i dati di log diventino disponibili per l'esportazione. Le attività di esportazione scadono dopo 24 ore. Se le tue attività di esportazione sono scadute, riduci l'intervallo di tempo quando crei l'attività di esportazione.

Per un'analisi quasi in tempo reale dei dati di log, consulta Analisi dei dati di registro con CloudWatch Logs Insights o Elaborazione in tempo reale dei dati di log con le sottoscrizioni.

Indice

Concetti

Prima di iniziare, acquisisci familiarità con i seguenti concetti di esportazione:

nome gruppo di log

Il nome del gruppo di log associato a un'attività di esportazione. I dati di log di questo gruppo di log verranno esportati nel bucket S3 specificato.

da (timestamp)

Un timestamp obbligatorio espresso come il numero di millisecondi a partire dal 1° gennaio 1970 00:00:00 UTC. Tutti gli eventi di log nel gruppo di log inseriti dopo questo momento saranno esportati.

a (timestamp)

Un timestamp obbligatorio espresso come il numero di millisecondi a partire dal 1° gennaio 1970 00:00:00 UTC. Tutti gli eventi di log nel gruppo di log inseriti prima di questo momento saranno esportati.

bucket di destinazione

Il nome del bucket S3 associato a un'attività di esportazione. Questo bucket viene utilizzato per esportare i dati di log dal gruppo di log specificato.

prefisso di destinazione

Un attributo facoltativo utilizzato come prefisso della chiave Amazon S3 per tutti gli oggetti esportati. Questo ti aiuta a creare un'organizzazione con stile cartella nel bucket.

Esportazione di dati di log in Amazon S3 tramite la console

Negli esempi seguenti, utilizzi la CloudWatch console Amazon per esportare tutti i dati da un gruppo di log di Amazon CloudWatch Logs denominato in un `my-log-group` bucket Amazon S3 denominato. `my-exported-logs`

L'esportazione dei dati di log in bucket S3 crittografati da SSE-KMS è supportata. L'esportazione in bucket crittografati con DSSE-KMS non è supportata.

I dettagli su come configuri l'esportazione dipendono dal fatto che il bucket Amazon S3 in cui desideri esportare si trovi nello stesso account dei log che vengono esportati o in un account diverso.

Argomenti

Esportazione nello stesso account

Se il bucket Amazon S3 si trova nello stesso account dei log che vengono esportati, segui le istruzioni in questa sezione.

Argomenti

- Passaggio 5: Creazione di un'attività di esportazione

Fase 1: creazione di un bucket Amazon S3

Ti consigliamo di utilizzare un bucket creato appositamente per Logs. CloudWatch Tuttavia, se intendi utilizzare un bucket esistente, puoi passare alla fase 2.

> **Note**
>
> Il bucket S3 deve risiedere nella stessa regione dei dati di registro da esportare. CloudWatch I registri non supportano l'esportazione di dati in bucket S3 in una regione diversa.

Per creare un bucket S3

1. Apri la console Amazon S3 all'indirizzo https://console.aws.amazon.com/s3/.
2. Se necessario, modificare la regione . Dalla barra di navigazione, scegli la regione in cui risiedono i registri. CloudWatch
3. Scegli Crea bucket.
4. In Bucket Name (Nome bucket), immettere il nome del bucket.
5. Per Regione, seleziona la regione in cui risiedono i dati CloudWatch dei registri.
6. Scegli Crea.

Fase 2: configurare le autorizzazioni di accesso

Per creare l'attività di esportazione nella fase 5, è necessario essere registrati con il ruolo IAM `AmazonS3ReadOnlyAccess` e con le seguenti autorizzazioni:

- `logs:CreateExportTask`
- `logs:CancelExportTask`
- `logs:DescribeExportTasks`
- `logs:DescribeLogStreams`
- `logs:DescribeLogGroups`

Per fornire l'accesso, aggiungi autorizzazioni ai tuoi utenti, gruppi o ruoli:

- Utenti e gruppi in: AWS IAM Identity Center

  Crea un set di autorizzazioni. Segui le istruzioni riportate nella pagina Create a permission set (Creazione di un set di autorizzazioni) nella Guida per l'utente diAWS IAM Identity Center .

- Utenti gestiti in IAM tramite un provider di identità:

  Crea un ruolo per la federazione delle identità. Segui le istruzioni riportate nella pagina Creating a role for a third-party identity provider (federation) (Creazione di un ruolo per un provider di identità di terze parti [federazione]) nella Guida per l'utente di IAM.

- Utenti IAM:
  - Crea un ruolo che l'utente possa assumere. Per istruzioni, consulta la pagina Creating a role for an IAM user (Creazione di un ruolo per un utente IAM) nella Guida per l'utente di IAM.
  - (Non consigliato) Collega una policy direttamente a un utente o aggiungi un utente a un gruppo di utenti. Segui le istruzioni riportate nella pagina Aggiunta di autorizzazioni a un utente (console) nella Guida per l'utente di IAM.

Passaggio 3: Impostazione delle autorizzazioni su un bucket S3

Per impostazione predefinita, tutti i bucket e gli oggetti S3 sono privati. Solo il proprietario della risorsa, l' Account AWS che ha creato il bucket, può accedere al bucket e agli oggetti in esso contenuti. Tuttavia, il proprietario della risorsa può concedere le autorizzazioni di accesso ad altre risorse e ad altri utenti mediante una policy di accesso.

Quando si imposta la policy, consigliamo di includere una stringa generata in modo casuale come prefisso per il bucket, in modo che solo i flussi di log vengano esportati nel bucket.

> **Important**
>
> Per rendere più sicure le esportazioni verso i bucket S3, ora ti chiediamo di specificare l'elenco degli account di origine autorizzati a esportare i dati di log nel tuo bucket S3. Nell'esempio seguente, l'elenco degli ID di account nella chiave `aws:SourceAccount` è costituito dagli account da cui un utente può esportare i dati di log nel bucket S3. La chiave `aws:SourceArn` è la risorsa per la quale viene intrapresa l'azione. Puoi limitarla a un gruppo di log specifico o utilizzare un jolly come mostrato in questo esempio.
> Ti consigliamo di includere anche l'ID dell'account in cui è stato creato il bucket S3 per consentire l'esportazione all'interno dello stesso account.

Impostazione delle autorizzazioni su un bucket Amazon S3

1. Nella console Amazon S3, scegliere il bucket creato nella fase 1.
2. Selezionare Permissions (Autorizzazioni), Add bucket policy (Aggiungi policy bucket).
3. In Bucket Policy Editor (Editor della policy del bucket), aggiungi la seguente policy. Cambia `my-exported-logs` nel nome del bucket S3. Assicurati di specificare l'endpoint corretto della regione, come `us-west-1`, per Principale.

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
          "Action": "s3:GetBucketAcl",
          "Effect": "Allow",
          "Resource": "arn:aws:s3:::my-exported-logs",
          "Principal": { "Service": "logs.Region.amazonaws.com" },
          "Condition": {
            "StringEquals": {
                "aws:SourceAccount": [
                    "AccountId1",
                    "AccountId2",
                    ...
                ]
            },
            "ArnLike": {
                    "aws:SourceArn": [
                        "arn:aws:logs:Region:AccountId1:log-group:*",
                        "arn:aws:logs:Region:AccountId2:log-group:*",
                        ...
                     ]
            }
          }
      },
      {
          "Action": "s3:PutObject" ,
          "Effect": "Allow",
          "Resource": "arn:aws:s3:::my-exported-logs/*",
          "Principal": { "Service": "logs.Region.amazonaws.com" },
          "Condition": {
            "StringEquals": {
                "s3:x-amz-acl": "bucket-owner-full-control",
                "aws:SourceAccount": [
```

```
                    "AccountId1",
                    "AccountId2",
                    ...
                ]
            },
            "ArnLike": {
                    "aws:SourceArn": [
                        "arn:aws:logs:Region:AccountId1:log-group:*",
                        "arn:aws:logs:Region:AccountId2:log-group:*",
                        ...
                    ]
            }
          }
      }
    ]
}
```

4. Seleziona Save (Salva) per impostare la policy appena aggiunta come la policy di accesso all'interno del bucket. Questa politica consente a CloudWatch Logs di esportare i dati di registro nel bucket S3. Il proprietario del bucket dispone di autorizzazioni complete su tutti gli oggetti esportati.

> **Warning**
>
> Se al bucket esistente sono già associate una o più politiche, aggiungi le istruzioni per CloudWatch Logs access a quella o più policy. Consigliamo di valutare il set di autorizzazioni risultante per verificare che siano adeguate agli utenti che accederanno al bucket.

(Opzionale) Passaggio 4: Esportazione in un bucket crittografato con SSE-KMS

Questo passaggio è necessario solo se si esegue l'esportazione in un bucket S3 che utilizza la crittografia lato server con. AWS KMS keysQuesta crittografia è nota come SSE-KMS.

Esportazione di un bucket crittografato con SSE-KMS

1. Apri la console all'indirizzo https://console.aws.amazon.com/kms. AWS KMS
2. Per modificare la Regione AWS, usa il selettore della regione nell'angolo superiore destro della pagina.

3. Nella barra di navigazione a sinistra, scegli Customer managed keys (Chiavi gestite dal cliente).

   Scegli Create Key (Crea chiave).
4. Alla voce Key type (Tipo di chiave), scegliere Symmetric (Simmetrica).
5. Per Key usage (Utilizzo della chiave), scegli Encrypt and decrypt (Crittografa e decrittografa), quindi scegli Next (Avanti).
6. In Add labels (Aggiungi etichette), inserisci un alias per la chiave e, facoltativamente, aggiungi una descrizione o dei tag. Quindi scegli Successivo.
7. In Key administrators (Amministratori delle chiavi), seleziona chi può amministrare questa chiave, quindi scegli Next (Avanti).
8. In Define key usage permissions (Definisci le autorizzazioni per utilizzare la chiave), non apportare modifiche e scegli Next (Avanti).
9. Esamina le impostazioni e scegli Finish (Fine).
10. Torna alla pagina Customer managed keys (Chiavi gestite dal cliente) e scegli il nome della chiave che hai appena creato.
11. Nella sezione Key policy (Policy chiave), scegli Switch to policy view (Passa alla visualizzazione della policy).
12. Nella sezione Key policy (Policy chiave), scegli Edit (Modifica).
13. Aggiungi la seguente istruzione all'elenco delle istruzioni della policy chiave. Quando lo fai, sostituisci i campi *Region* con la Regione dei tuoi log e *account-ARN* con l'ARN dell'account proprietario della chiave KMS.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Allow CWL Service Principal usage",
            "Effect": "Allow",
            "Principal": {
                "Service": "logs.Region.amazonaws.com"
            },
            "Action": [
                "kms:GenerateDataKey",
                "kms:Decrypt"
            ],
            "Resource": "*"
        },
```

```
        {
            "Sid": "Enable IAM User Permissions",
            "Effect": "Allow",
            "Principal": {
                "AWS": "account-ARN"
            },
            "Action": [
                "kms:GetKeyPolicy*",
                "kms:PutKeyPolicy*",
                "kms:DescribeKey*",
                "kms:CreateAlias*",
                "kms:ScheduleKeyDeletion*",
                "kms:Decrypt"
            ],
            "Resource": "*"
        }
    ]
}
```

14. Seleziona Salvataggio delle modifiche.
15. Apri la console Amazon S3 all'indirizzo https://console.aws.amazon.com/s3/.
16. Cerca il bucket che hai creato in Fase 1: Creazione di un bucket S3 e scegli il nome del bucket.
17. Scegliere la scheda Properties (Proprietà). Quindi, in Default Encryption (Crittografia predefinita), scegli Edit (Modifica).
18. In Server-side Encryption (Crittografia lato server), scegli Enable (Abilita).
19. In Tipo di crittografia, scegli ChiaveAWS Key Management Service (SSE-KMS).
20. Scegli tra AWS KMS le tue chiavi e trova la chiave che hai creato.
21. Per Bucket key (Chiave bucket), scegli Enable (Abilita).
22. Seleziona Salvataggio delle modifiche.

Passaggio 5: Creazione di un'attività di esportazione

In questa fase, verrà creata l'attività di esportazione per esportare log da un gruppo di log.

Per esportare dati su Amazon S3 utilizzando la console CloudWatch

1. Accedi con le autorizzazioni sufficienti come documentato in Fase 2: configurare le autorizzazioni di accesso.
2. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.

3. Nel pannello di navigazione, selezionare Log groups (Gruppi di log).
4. Nella schermata Gruppi di log, scegliere il nome del gruppo di log.
5. Scegli Actions (Operazioni), Export data to Amazon S3 (Esporta dati in Amazon S3.
6. Nella schermata Export data to Amazon S3 (Esporta dati in Amazon S3), in Define data export (Definizione dell'esportazione dei dati), impostare l'intervallo di tempo per i dati da esportare in From (Da) e To (A).
7. Se il gruppo di log dispone di più flussi di log, puoi fornire un prefisso del flusso di log per limitare i dati del gruppo di log a un flusso specifico. Scegliere Advanced (Avanzato) e immettere il prefisso del flusso di log in Stream prefix (Prefisso flusso).
8. In Choose S3 bucket (Scegli bucket S3), seleziona l'account associato al bucket S3.
9. In S3 bucket name (Nome bucket S3), seleziona un bucket S3.
10. Per Prefisso bucket S3, immettere la stringa generata in modo casuale specificata nella policy del bucket.
11. Seleziona Esporta per esportare i dati di log in Amazon S3.
12. Per visualizzare lo stato dei dati di log esportati in Amazon S3, scegli Operazioni, quindi Visualizza tutte le esportazioni in Amazon S3.

Esportazione in account diversi

Se il bucket Amazon S3 si trova in un account diverso da quello dei log che vengono esportati, segui le istruzioni in questa sezione.

Argomenti

Fase 1: creazione di un bucket Amazon S3

Ti consigliamo di utilizzare un bucket creato appositamente per i CloudWatch log. Tuttavia, se intendi utilizzare un bucket esistente, puoi passare alla fase 2.

> **Note**
>
> Il bucket S3 deve risiedere nella stessa regione dei dati di registro da esportare. CloudWatch I registri non supportano l'esportazione di dati in bucket S3 in una regione diversa.

Per creare un bucket S3

1. Apri la console Amazon S3 all'indirizzo https://console.aws.amazon.com/s3/.
2. Se necessario, modificare la regione . Dalla barra di navigazione, scegli la regione in cui risiedono i registri. CloudWatch
3. Scegli Crea bucket.
4. In Bucket Name (Nome bucket), immettere il nome del bucket.
5. Per Regione, seleziona la regione in cui risiedono i dati CloudWatch dei registri.
6. Scegli Crea.

Fase 2: configurare le autorizzazioni di accesso

Innanzitutto, devi creare una nuova policy IAM per consentire a CloudWatch Logs di avere l'`s3:PutObject`autorizzazione per il bucket Amazon S3 di destinazione nell'account di destinazione.

La policy che crei dipende dal fatto che il bucket di destinazione utilizzi la crittografia. AWS KMS

Creazione di una policy IAM per esportare i log in un bucket Amazon S3

1. Aprire la console IAM all'indirizzo https://console.aws.amazon.com/iam/.
2. Nel riquadro di navigazione a sinistra, seleziona Policies (Policy).
3. Scegli Crea policy.
4. Nella sezione Editor di policy, scegli l'opzione JSON.
5. Se il bucket di destinazione non utilizza AWS KMS la crittografia, incolla la seguente politica nell'editor.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
```

```
            "Effect": "Allow",
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::my-exported-logs/*"
        }
    ]
}
```

Se il bucket di destinazione utilizza la AWS KMS crittografia, incolla la seguente politica nell'editor.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "s3:PutObject",
      "Resource": "arn:aws:s3:::my-exported-logs/*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "kms:GenerateDataKey",
        "kms:Decrypt"
      ],
      "Resource": "ARN_OF_KMS_KEY"
    }
  ]
}
```

6. Seleziona Avanti.
7. Inserire un nome per la policy. Utilizzerai questo nome per collegare la policy al tuo ruolo IAM.
8. Quindi, per salvare la nuova policy, seleziona Crea policy.

Per creare l'attività di esportazione nella fase 5, sarà necessario essersi registrati con il ruolo IAM `AmazonS3ReadOnlyAccess`. Devi inoltre avere effettuato l'accesso con la policy IAM appena creata e anche con le seguenti autorizzazioni:

- `logs:CreateExportTask`
- `logs:CancelExportTask`
- `logs:DescribeExportTasks`

- `logs:DescribeLogStreams`
- `logs:DescribeLogGroups`

Per fornire l'accesso, aggiungi autorizzazioni ai tuoi utenti, gruppi o ruoli:

- Utenti e gruppi in AWS IAM Identity Center:

  Crea un set di autorizzazioni. Segui le istruzioni riportate nella pagina Create a permission set (Creazione di un set di autorizzazioni) nella Guida per l'utente diAWS IAM Identity Center .
- Utenti gestiti in IAM tramite un provider di identità:

  Crea un ruolo per la federazione delle identità. Segui le istruzioni riportate nella pagina Creating a role for a third-party identity provider (federation) (Creazione di un ruolo per un provider di identità di terze parti [federazione]) nella Guida per l'utente di IAM.
- Utenti IAM:
  - Crea un ruolo che l'utente possa assumere. Per istruzioni, consulta la pagina Creating a role for an IAM user (Creazione di un ruolo per un utente IAM) nella Guida per l'utente di IAM.
  - (Non consigliato) Collega una policy direttamente a un utente o aggiungi un utente a un gruppo di utenti. Segui le istruzioni riportate nella pagina Aggiunta di autorizzazioni a un utente (console) nella Guida per l'utente di IAM.

Passaggio 3: Impostazione delle autorizzazioni su un bucket S3

Per impostazione predefinita, tutti i bucket e gli oggetti S3 sono privati. Solo il proprietario della risorsa, l' Account AWS che ha creato il bucket, può accedere al bucket e agli oggetti in esso contenuti. Tuttavia, il proprietario della risorsa può concedere le autorizzazioni di accesso ad altre risorse e ad altri utenti mediante una policy di accesso.

Quando si imposta la policy, consigliamo di includere una stringa generata in modo casuale come prefisso per il bucket, in modo che solo i flussi di log vengano esportati nel bucket.

> **Important**
>
> Per rendere più sicure le esportazioni verso i bucket S3, ora ti chiediamo di specificare l'elenco degli account di origine autorizzati a esportare i dati di log nel tuo bucket S3. Nell'esempio seguente, l'elenco degli ID di account nella chiave `aws:SourceAccount` è costituito dagli account da cui un utente può esportare i dati di log nel bucket S3. La chiave

`aws:SourceArn` è la risorsa per la quale viene intrapresa l'azione. Puoi limitarla a un gruppo di log specifico o utilizzare un jolly come mostrato in questo esempio.
Ti consigliamo di includere anche l'ID dell'account in cui è stato creato il bucket S3 per consentire l'esportazione all'interno dello stesso account.

Impostazione delle autorizzazioni su un bucket Amazon S3

1. Nella console Amazon S3, scegliere il bucket creato nella fase 1.
2. Selezionare Permissions (Autorizzazioni), Add bucket policy (Aggiungi policy bucket).
3. In Bucket Policy Editor (Editor della policy del bucket), aggiungi la seguente policy. Cambia `my-exported-logs` nel nome del bucket S3. Assicurati di specificare l'endpoint corretto della regione, come `us-west-1`, per Principale.

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
          "Action": "s3:GetBucketAcl",
          "Effect": "Allow",
          "Resource": "arn:aws:s3:::my-exported-logs",
          "Principal": { "Service": "logs.Region.amazonaws.com" },
          "Condition": {
            "StringEquals": {
                "aws:SourceAccount": [
                    "AccountId1",
                    "AccountId2",
                    ...
                ]
            },
            "ArnLike": {
                    "aws:SourceArn": [
                        "arn:aws:logs:Region:AccountId1:log-group:*",
                        "arn:aws:logs:Region:AccountId2:log-group:*",
                        ...
                     ]
            }
          }
      },
      {
          "Action": "s3:PutObject" ,
```

```
            "Effect": "Allow",
            "Resource": "arn:aws:s3:::my-exported-logs/*",
            "Principal": { "Service": "logs.Region.amazonaws.com" },
            "Condition": {
              "StringEquals": {
                  "s3:x-amz-acl": "bucket-owner-full-control",
                  "aws:SourceAccount": [
                      "AccountId1",
                      "AccountId2",
                      ...
                  ]
              },
              "ArnLike": {
                      "aws:SourceArn": [
                          "arn:aws:logs:Region:AccountId1:log-group:*",
                          "arn:aws:logs:Region:AccountId2:log-group:*",
                          ...
                      ]
              }
            }
        },
        {
            "Effect": "Allow",
            "Principal": {
              "AWS": "arn:aws:iam::create_export_task_caller_account:role/role_name"
            },
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::my-exported-logs/*",
            "Condition": {
              "StringEquals": {
                  "s3:x-amz-acl": "bucket-owner-full-control"
              }
            }
         }
    ]
}
```

4. Seleziona Save (Salva) per impostare la policy appena aggiunta come la policy di accesso all'interno del bucket. Questa politica consente a CloudWatch Logs di esportare i dati di registro nel bucket S3. Il proprietario del bucket dispone di autorizzazioni complete su tutti gli oggetti esportati.

> **Warning**
>
> Se al bucket esistente sono già associate una o più politiche, aggiungi le istruzioni per CloudWatch Logs access a quella o più policy. Consigliamo di valutare il set di autorizzazioni risultante per verificare che siano adeguate agli utenti che accederanno al bucket.

(Opzionale) Passaggio 4: Esportazione in un bucket crittografato con SSE-KMS

Questo passaggio è necessario solo se si esegue l'esportazione in un bucket S3 che utilizza la crittografia lato server con. AWS KMS keysQuesta crittografia è nota come SSE-KMS.

Esportazione di un bucket crittografato con SSE-KMS

1. Apri la console all'indirizzo https://console.aws.amazon.com/kms. AWS KMS
2. Per modificare la Regione AWS, usa il selettore della regione nell'angolo superiore destro della pagina.
3. Nella barra di navigazione a sinistra, scegli Customer managed keys (Chiavi gestite dal cliente).

   Scegli Create Key (Crea chiave).
4. Alla voce Key type (Tipo di chiave), scegliere Symmetric (Simmetrica).
5. Per Key usage (Utilizzo della chiave), scegli Encrypt and decrypt (Crittografa e decrittografa), quindi scegli Next (Avanti).
6. In Add labels (Aggiungi etichette), inserisci un alias per la chiave e, facoltativamente, aggiungi una descrizione o dei tag. Quindi scegli Successivo.
7. In Key administrators (Amministratori delle chiavi), seleziona chi può amministrare questa chiave, quindi scegli Next (Avanti).
8. In Define key usage permissions (Definisci le autorizzazioni per utilizzare la chiave), non apportare modifiche e scegli Next (Avanti).
9. Esamina le impostazioni e scegli Finish (Fine).
10. Torna alla pagina Customer managed keys (Chiavi gestite dal cliente) e scegli il nome della chiave che hai appena creato.
11. Nella sezione Key policy (Policy chiave), scegli Switch to policy view (Passa alla visualizzazione della policy).

12. Nella sezione Key policy (Policy chiave), scegli Edit (Modifica).
13. Aggiungi la seguente istruzione all'elenco delle istruzioni della policy chiave. Quando lo fai, sostituisci i campi *Region* con la Regione dei tuoi log e *account-ARN* con l'ARN dell'account proprietario della chiave KMS.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Allow CWL Service Principal usage",
            "Effect": "Allow",
            "Principal": {
                "Service": "logs.Region.amazonaws.com"
            },
            "Action": [
                "kms:GenerateDataKey",
                "kms:Decrypt"
            ],
            "Resource": "*"
        },
        {
            "Sid": "Enable IAM User Permissions",
            "Effect": "Allow",
            "Principal": {
                "AWS": "account-ARN"
            },
            "Action": [
                "kms:GetKeyPolicy*",
                "kms:PutKeyPolicy*",
                "kms:DescribeKey*",
                "kms:CreateAlias*",
                "kms:ScheduleKeyDeletion*",
                "kms:Decrypt"
            ],
            "Resource": "*"
        },
        {
            "Sid": "Enable IAM Role Permissions",
            "Effect": "Allow",
            "Principal": {
                "AWS":
 "arn:aws:iam::create_export_task_caller_account:role/role_name"
            },
```

```
            "Action": [
                "kms:GenerateDataKey",
                "kms:Decrypt"
            ],
            "Resource": "ARN_OF_KMS_KEY"
        }
    ]
}
```

14. Seleziona Salvataggio delle modifiche.
15. Apri la console Amazon S3 all'indirizzo https://console.aws.amazon.com/s3/.
16. Cerca il bucket che hai creato in Fase 1: Creazione di un bucket S3 e scegli il nome del bucket.
17. Scegliere la scheda Properties (Proprietà). Quindi, in Default Encryption (Crittografia predefinita), scegli Edit (Modifica).
18. In Server-side Encryption (Crittografia lato server), scegli Enable (Abilita).
19. In Tipo di crittografia, scegli ChiaveAWS Key Management Service (SSE-KMS).
20. Scegli tra AWS KMS le tue chiavi e trova la chiave che hai creato.
21. Per Bucket key (Chiave bucket), scegli Enable (Abilita).
22. Seleziona Salvataggio delle modifiche.

Passaggio 5: Creazione di un'attività di esportazione

In questa fase, verrà creata l'attività di esportazione per esportare log da un gruppo di log.

Per esportare dati su Amazon S3 utilizzando la console CloudWatch

1. Accedi con le autorizzazioni sufficienti come documentato in Fase 2: configurare le autorizzazioni di accesso.
2. Apri la CloudWatch console all'indirizzo https://console.aws.amazon.com/cloudwatch/.
3. Nel pannello di navigazione, selezionare Log groups (Gruppi di log).
4. Nella schermata Gruppi di log, scegliere il nome del gruppo di log.
5. Scegli Actions (Operazioni), Export data to Amazon S3 (Esporta dati in Amazon S3.
6. Nella schermata Export data to Amazon S3 (Esporta dati in Amazon S3), in Define data export (Definizione dell'esportazione dei dati), impostare l'intervallo di tempo per i dati da esportare in From (Da) e To (A).

7. Se il gruppo di log dispone di più flussi di log, puoi fornire un prefisso del flusso di log per limitare i dati del gruppo di log a un flusso specifico. Scegliere Advanced (Avanzato) e immettere il prefisso del flusso di log in Stream prefix (Prefisso flusso).
8. In Choose S3 bucket (Scegli bucket S3), seleziona l'account associato al bucket S3.
9. In S3 bucket name (Nome bucket S3), seleziona un bucket S3.
10. Per Prefisso bucket S3, immettere la stringa generata in modo casuale specificata nella policy del bucket.
11. Seleziona Esporta per esportare i dati di log in Amazon S3.
12. Per visualizzare lo stato dei dati di log esportati in Amazon S3, scegli Operazioni, quindi Visualizza tutte le esportazioni in Amazon S3.

Esporta i dati di registro su Amazon S3 utilizzando il AWS CLI

Nell'esempio seguente, si utilizza un'attività di esportazione per esportare tutti i dati da un gruppo di log CloudWatch Logs denominato in un `my-log-group` bucket Amazon S3 denominato. `my-exported-logs` Questo esempio presuppone che tu abbia già creato il gruppo di log `my-log-group`.

È supportata l'esportazione dei dati di registro in bucket S3 crittografati da. AWS KMS L'esportazione in bucket crittografati con DSSE-KMS non è supportata.

I dettagli su come configuri l'esportazione dipendono dal fatto che il bucket Amazon S3 in cui desideri esportare si trovi nello stesso account dei log che vengono esportati o in un account diverso.

Argomenti

- Esportazione nello stesso account
- Esportazione in account diversi

Esportazione nello stesso account

Se il bucket Amazon S3 si trova nello stesso account dei log che vengono esportati, segui le istruzioni in questa sezione.

Argomenti

- Fase 1: Creazione di un bucket S3

Fase 1: Creazione di un bucket S3

Ti consigliamo di utilizzare un bucket creato appositamente per i log. CloudWatch Tuttavia, se intendi utilizzare un bucket esistente, puoi passare alla fase 2.

> **Note**
>
> Il bucket S3 deve risiedere nella stessa regione dei dati di registro da esportare. CloudWatch I registri non supportano l'esportazione di dati in bucket S3 in una regione diversa.

Per creare un bucket S3 utilizzando il AWS CLI

Al prompt dei comandi, eseguire il seguente comando create-bucket, in cui `LocationConstraint` è la regione in cui esportare i dati di log.

```
aws s3api create-bucket --bucket my-exported-logs --create-bucket-configuration
 LocationConstraint=us-east-2
```

Di seguito è riportato un output di esempio.

```
{
    "Location": "/my-exported-logs"
}
```

Fase 2: configurare le autorizzazioni di accesso

Per creare l'attività di esportazione nella fase 5, è necessario essere registrati con il ruolo IAM `AmazonS3ReadOnlyAccess` e con le seguenti autorizzazioni:

- `logs:CreateExportTask`
- `logs:CancelExportTask`
- `logs:DescribeExportTasks`

- `logs:DescribeLogStreams`
- `logs:DescribeLogGroups`

Per fornire l'accesso, aggiungi autorizzazioni ai tuoi utenti, gruppi o ruoli:

- Utenti e gruppi in: AWS IAM Identity Center

  Crea un set di autorizzazioni. Segui le istruzioni riportate nella pagina Create a permission set (Creazione di un set di autorizzazioni) nella Guida per l'utente diAWS IAM Identity Center .

- Utenti gestiti in IAM tramite un provider di identità:

  Crea un ruolo per la federazione delle identità. Segui le istruzioni riportate nella pagina Creating a role for a third-party identity provider (federation) (Creazione di un ruolo per un provider di identità di terze parti [federazione]) nella Guida per l'utente di IAM.

- Utenti IAM:
  - Crea un ruolo che l'utente possa assumere. Per istruzioni, consulta la pagina Creating a role for an IAM user (Creazione di un ruolo per un utente IAM) nella Guida per l'utente di IAM.
  - (Non consigliato) Collega una policy direttamente a un utente o aggiungi un utente a un gruppo di utenti. Segui le istruzioni riportate nella pagina Aggiunta di autorizzazioni a un utente (console) nella Guida per l'utente di IAM.

Passaggio 3: Impostazione delle autorizzazioni su un bucket S3

Per impostazione predefinita, tutti i bucket e gli oggetti S3 sono privati. Solo il proprietario della risorsa, l'account che ha creato il bucket, può accedere al bucket e agli oggetti in esso contenuti. Tuttavia, il proprietario della risorsa può concedere le autorizzazioni di accesso ad altre risorse e ad altri utenti mediante una policy di accesso.

> **Important**
>
> Per rendere più sicure le esportazioni verso i bucket S3, ora ti chiediamo di specificare l'elenco degli account di origine autorizzati a esportare i dati di log nel tuo bucket S3. Nell'esempio seguente, l'elenco degli ID di account nella chiave `aws:SourceAccount` è costituito dagli account da cui un utente può esportare i dati di log nel bucket S3. La chiave `aws:SourceArn` è la risorsa per la quale viene intrapresa l'azione. Puoi limitarla a un gruppo di log specifico o utilizzare un jolly come mostrato in questo esempio.

Ti consigliamo di includere anche l'ID dell'account in cui è stato creato il bucket S3 per consentire l'esportazione all'interno dello stesso account.

Impostazione delle autorizzazioni su un bucket S3

1. Crea il file `policy.json` e aggiungi la seguente policy di accesso, modificando `my-exported-logs` con il nome del bucket S3 e `Principal` con l'endpoint della regione di esportazione dei dati di log, come `us-west-1`. Utilizza un editor di testo per creare questo file di policy. Non utilizzare la console IAM.

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
          "Action": "s3:GetBucketAcl",
          "Effect": "Allow",
          "Resource": "arn:aws:s3:::my-exported-logs",
          "Principal": { "Service": "logs.Region.amazonaws.com" },
          "Condition": {
            "StringEquals": {
                "aws:SourceAccount": [
                    "AccountId1",
                    "AccountId2",
                    ...
                ]
            },
            "ArnLike": {
                    "aws:SourceArn": [
                        "arn:aws:logs:Region:AccountId1:log-group:*",
                        "arn:aws:logs:Region:AccountId2:log-group:*",
                        ...
                     ]
            }
          }
      },
      {
          "Action": "s3:PutObject" ,
          "Effect": "Allow",
          "Resource": "arn:aws:s3:::my-exported-logs/*",
          "Principal": { "Service": "logs.Region.amazonaws.com" },
```

```
            "Condition": {
              "StringEquals": {
                  "s3:x-amz-acl": "bucket-owner-full-control",
                  "aws:SourceAccount": [
                      "AccountId1",
                      "AccountId2",
                      ...
                  ]
              },
              "ArnLike": {
                      "aws:SourceArn": [
                          "arn:aws:logs:Region:AccountId1:log-group:*",
                          "arn:aws:logs:Region:AccountId2:log-group:*",
                          ...
                      ]
              }
            }
        }
    ]
}
```

2. Imposta la politica che hai appena aggiunto come politica di accesso nel tuo bucket utilizzando il put-bucket-policycomando. Questa politica consente a CloudWatch Logs di esportare i dati di registro nel bucket S3. Il proprietario del bucket disporrà di autorizzazioni complete su tutti gli oggetti esportati.

```
aws s3api put-bucket-policy --bucket my-exported-logs --policy file://policy.json
```

> **Warning**
>
> Se al bucket esistente sono già associate una o più politiche, aggiungi le istruzioni per CloudWatch Logs access a quella o più policy. Consigliamo di valutare il set di autorizzazioni risultante per verificare che siano adeguate agli utenti che accederanno al bucket.

(Opzionale) Passaggio 4: Esportazione in un bucket crittografato con SSE-KMS

Questo passaggio è necessario solo se si esegue l'esportazione in un bucket S3 che utilizza la crittografia lato server con. AWS KMS keysQuesta crittografia è nota come SSE-KMS.

Esportazione di un bucket crittografato con SSE-KMS

1. Utilizza un editor di testo per creare un file denominato `key_policy.json` e aggiungi la seguente policy di accesso. Quando aggiungi la policy della chiave, apporta le modifiche seguenti:

   - Sostituisci il campo *`Region`* con la Regione dei tuoi log.
   - Sostituisci il campo *`account-ARN`* con l'ARN dell'account proprietario della chiave KMS.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Allow CWL Service Principal usage",
            "Effect": "Allow",
            "Principal": {
                "Service": "logs.Region.amazonaws.com"
            },
            "Action": [
                "kms:GenerateDataKey",
                "kms:Decrypt"
            ],
            "Resource": "*"
        },
        {
            "Sid": "Enable IAM User Permissions",
            "Effect": "Allow",
            "Principal": {
                "AWS": "account-ARN"
            },
            "Action": [
                "kms:GetKeyPolicy*",
                "kms:PutKeyPolicy*",
                "kms:DescribeKey*",
                "kms:CreateAlias*",
                "kms:ScheduleKeyDeletion*",
                "kms:Decrypt"
            ],
            "Resource": "*"
        }
    ]
```

```
}
```

2. Immetti il comando seguente:

```
aws kms create-key --policy file://key_policy.json
```

Di seguito è fornito un esempio dell'output di questo comando:

```
{
    "KeyMetadata": {
        "AWSAccountId": "account_id",
        "KeyId": "key_id",
        "Arn": "arn:aws:kms:us-east-2:account_id:key/key_id",
        "CreationDate": "time",
        "Enabled": true,
        "Description": "",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "KeyState": "Enabled",
        "Origin": "AWS_KMS",
        "KeyManager": "CUSTOMER",
        "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "KeySpec": "SYMMETRIC_DEFAULT",
        "EncryptionAlgorithms": [
            "SYMMETRIC_DEFAULT"
        ],
        "MultiRegion": false
    }
```

3. Utilizza un editor di testo per creare un file denominato `bucketencryption.json` con il seguente contenuto.

```
{
  "Rules": [
    {
      "ApplyServerSideEncryptionByDefault": {
        "SSEAlgorithm": "aws:kms",
        "KMSMasterKeyID": "{KMS Key ARN}"
      },
      "BucketKeyEnabled": true
    }
  ]
}
```

4. Inserisci il comando seguente, sostituendo *bucket-name* con il nome del bucket nel quale vuoi esportare i log.

```
aws s3api put-bucket-encryption --bucket bucket-name --server-side-encryption-
configuration file://bucketencryption.json
```

Se il comando non restituisce un errore, il processo ha esito positivo.

Passaggio 5: Creazione di un'attività di esportazione

Usare il comando seguente per creare l'attività di esportazione. Dopo averla creata, potrebbero essere necessari da pochi secondi ad alcune ore per l'attività di esportazione, in base alla dimensione dei dati da esportare.

Per esportare dati in Amazon S3 utilizzando il AWS CLI

1. Accedi con le autorizzazioni sufficienti come documentato in Fase 2: configurare le autorizzazioni di accesso.
2. Al prompt dei comandi, usa il seguente create-export-taskcomando per creare l'attività di esportazione.

```
aws logs create-export-task --profile CWLExportUser --task-name "my-log-
group-09-10-2015" --log-group-name "my-log-group" --from 1441490400000 --
to 1441494000000 --destination "my-exported-logs" --destination-prefix "export-
task-output"
```

Di seguito è riportato un output di esempio.

```
{
    "taskId": "cda45419-90ea-4db5-9833-aade86253e66"
}
```

Esportazione in account diversi

Se il bucket Amazon S3 si trova in un account diverso da quello dei log che vengono esportati, segui le istruzioni in questa sezione.

Argomenti

- Fase 1: Creazione di un bucket S3
- Fase 2: configurare le autorizzazioni di accesso
- Passaggio 3: Impostazione delle autorizzazioni su un bucket S3
- (Opzionale) Passaggio 4: Esportazione in un bucket crittografato con SSE-KMS
- Passaggio 5: Creazione di un'attività di esportazione

Fase 1: Creazione di un bucket S3

Si consiglia di utilizzare un bucket creato appositamente per CloudWatch i registri. Tuttavia, se intendi utilizzare un bucket esistente, puoi passare alla fase 2.

> **Note**
>
> Il bucket S3 deve risiedere nella stessa regione dei dati di registro da esportare. CloudWatch I registri non supportano l'esportazione di dati in bucket S3 in una regione diversa.

Per creare un bucket S3 utilizzando il AWS CLI

Al prompt dei comandi, eseguire il seguente comando create-bucket, in cui `LocationConstraint` è la regione in cui esportare i dati di log.

```
aws s3api create-bucket --bucket my-exported-logs --create-bucket-configuration
 LocationConstraint=us-east-2
```

Di seguito è riportato un output di esempio.

```
{
    "Location": "/my-exported-logs"
}
```

Fase 2: configurare le autorizzazioni di accesso

Innanzitutto, devi creare una nuova policy IAM per consentire a CloudWatch Logs di avere l'`s3:PutObject`autorizzazione per il bucket Amazon S3 di destinazione.

Per creare l'attività di esportazione nella fase 5, sarà necessario essersi registrati con il ruolo IAM `AmazonS3ReadOnlyAccess` e altre autorizzazioni. È possibile creare una policy che contenga alcune di queste altre autorizzazioni necessarie.

La policy che crei dipende dal fatto che il bucket di destinazione utilizzi la crittografia. AWS KMS Se non utilizza la AWS KMS crittografia, crea una politica con i seguenti contenuti.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::my-exported-logs/*"
        }
    ]
}
```

Se il bucket di destinazione utilizza AWS KMS la crittografia, create una policy con i seguenti contenuti.

```
{
    "Version": "2012-10-17",
    "Statement": [{
            "Effect": "Allow",
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::my-exported-logs/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:GenerateDataKey",
                "kms:Decrypt"
            ],
            "Resource": "ARN_OF_KMS_KEY"
        }
    ]
}
```

Per creare l'attività di esportazione nella fase 5, devi avere effettuato l'accesso con il ruolo IAM `AmazonS3ReadOnlyAccess`, la policy IAM appena creata e anche con le seguenti autorizzazioni:

- `logs:CreateExportTask`
- `logs:CancelExportTask`
- `logs:DescribeExportTasks`
- `logs:DescribeLogStreams`
- `logs:DescribeLogGroups`

Per fornire l'accesso, aggiungi autorizzazioni ai tuoi utenti, gruppi o ruoli:

- Utenti e gruppi in AWS IAM Identity Center:

  Crea un set di autorizzazioni. Segui le istruzioni riportate nella pagina Create a permission set (Creazione di un set di autorizzazioni) nella Guida per l'utente diAWS IAM Identity Center .

- Utenti gestiti in IAM tramite un provider di identità:

  Crea un ruolo per la federazione delle identità. Segui le istruzioni riportate nella pagina Creating a role for a third-party identity provider (federation) (Creazione di un ruolo per un provider di identità di terze parti [federazione]) nella Guida per l'utente di IAM.

- Utenti IAM:
  - Crea un ruolo che l'utente possa assumere. Per istruzioni, consulta la pagina Creating a role for an IAM user (Creazione di un ruolo per un utente IAM) nella Guida per l'utente di IAM.
  - (Non consigliato) Collega una policy direttamente a un utente o aggiungi un utente a un gruppo di utenti. Segui le istruzioni riportate nella pagina Aggiunta di autorizzazioni a un utente (console) nella Guida per l'utente di IAM.

Passaggio 3: Impostazione delle autorizzazioni su un bucket S3

Per impostazione predefinita, tutti i bucket e gli oggetti S3 sono privati. Solo il proprietario della risorsa, l'account che ha creato il bucket, può accedere al bucket e agli oggetti in esso contenuti. Tuttavia, il proprietario della risorsa può concedere le autorizzazioni di accesso ad altre risorse e ad altri utenti mediante una policy di accesso.

> **Important**
>
> Per rendere più sicure le esportazioni verso i bucket S3, ora ti chiediamo di specificare l'elenco degli account di origine autorizzati a esportare i dati di log nel tuo bucket S3.

Nell'esempio seguente, l'elenco degli ID di account nella chiave `aws:SourceAccount` è costituito dagli account da cui un utente può esportare i dati di log nel bucket S3. La chiave `aws:SourceArn` è la risorsa per la quale viene intrapresa l'azione. Puoi limitarla a un gruppo di log specifico o utilizzare un jolly come mostrato in questo esempio.
Ti consigliamo di includere anche l'ID dell'account in cui è stato creato il bucket S3 per consentire l'esportazione all'interno dello stesso account.

Impostazione delle autorizzazioni su un bucket S3

1. Crea il file `policy.json` e aggiungi la seguente policy di accesso, modificando `my-exported-logs` con il nome del bucket S3 e `Principal` con l'endpoint della regione di esportazione dei dati di log, come `us-west-1`. Utilizza un editor di testo per creare questo file di policy. Non utilizzare la console IAM.

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
          "Action": "s3:GetBucketAcl",
          "Effect": "Allow",
          "Resource": "arn:aws:s3:::my-exported-logs",
          "Principal": { "Service": "logs.Region.amazonaws.com" },
          "Condition": {
            "StringEquals": {
                "aws:SourceAccount": [
                    "AccountId1",
                    "AccountId2",
                    ...
                ]
            },
            "ArnLike": {
                    "aws:SourceArn": [
                        "arn:aws:logs:Region:AccountId1:log-group:*",
                        "arn:aws:logs:Region:AccountId2:log-group:*",
                        ...
                     ]
            }
          }
      },
```

```
      {
          "Action": "s3:PutObject" ,
          "Effect": "Allow",
          "Resource": "arn:aws:s3:::my-exported-logs/*",
          "Principal": { "Service": "logs.Region.amazonaws.com" },
          "Condition": {
            "StringEquals": {
                "s3:x-amz-acl": "bucket-owner-full-control",
                "aws:SourceAccount": [
                    "AccountId1",
                    "AccountId2",
                    ...
                ]
            },
            "ArnLike": {
                    "aws:SourceArn": [
                        "arn:aws:logs:Region:AccountId1:log-group:*",
                        "arn:aws:logs:Region:AccountId2:log-group:*",
                        ...
                    ]
            }
          }
      },
      {
          "Effect": "Allow",
          "Principal": {
            "AWS": "arn:aws:iam::create_export_task_caller_account:role/role_name"
          },
          "Action": "s3:PutObject",
          "Resource": "arn:aws:s3:::my-exported-logs/*",
          "Condition": {
            "StringEquals": {
                "s3:x-amz-acl": "bucket-owner-full-control"
            }
          }
       }
    ]
}
```

2. Imposta la politica che hai appena aggiunto come politica di accesso nel tuo bucket utilizzando il put-bucket-policycomando. Questa politica consente a CloudWatch Logs di esportare i dati di registro nel bucket S3. Il proprietario del bucket disporrà di autorizzazioni complete su tutti gli oggetti esportati.

```
aws s3api put-bucket-policy --bucket my-exported-logs --policy file://policy.json
```

> **Warning**
>
> Se al bucket esistente sono già associate una o più politiche, aggiungi le istruzioni per CloudWatch Logs access a quella o più policy. Consigliamo di valutare il set di autorizzazioni risultante per verificare che siano adeguate agli utenti che accederanno al bucket.

(Opzionale) Passaggio 4: Esportazione in un bucket crittografato con SSE-KMS

Questo passaggio è necessario solo se si esegue l'esportazione in un bucket S3 che utilizza la crittografia lato server con. AWS KMS keysQuesta crittografia è nota come SSE-KMS.

Esportazione di un bucket crittografato con SSE-KMS

1. Utilizza un editor di testo per creare un file denominato `key_policy.json` e aggiungi la seguente policy di accesso. Quando aggiungi la policy della chiave, apporta le modifiche seguenti:

   - Sostituisci il campo *`Region`* con la Regione dei tuoi log.
   - Sostituisci il campo *`account-ARN`* con l'ARN dell'account proprietario della chiave KMS.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Allow CWL Service Principal usage",
            "Effect": "Allow",
            "Principal": {
                "Service": "logs.Region.amazonaws.com"
            },
            "Action": [
                "kms:GenerateDataKey",
                "kms:Decrypt"
            ],
            "Resource": "*"
```

```
        },
        {
            "Sid": "Enable IAM User Permissions",
            "Effect": "Allow",
            "Principal": {
                "AWS": "account-ARN"
            },
            "Action": [
                "kms:GetKeyPolicy*",
                "kms:PutKeyPolicy*",
                "kms:DescribeKey*",
                "kms:CreateAlias*",
                "kms:ScheduleKeyDeletion*",
                "kms:Decrypt"
            ],
            "Resource": "*"
        },
        {
            "Sid": "Enable IAM Role Permissions",
            "Effect": "Allow",
            "Principal": {
                "AWS":
 "arn:aws:iam::create_export_task_caller_account:role/role_name"
            },
            "Action": [
                "kms:GenerateDataKey",
                "kms:Decrypt"
            ],
            "Resource": "ARN_OF_KMS_KEY"
        }
    ]
}
```

2. Immetti il comando seguente:

```
aws kms create-key --policy file://key_policy.json
```

Di seguito è fornito un esempio dell'output di questo comando:

```
{
    "KeyMetadata": {
        "AWSAccountId": "account_id",
        "KeyId": "key_id",
```

```
        "Arn": "arn:aws:kms:us-east-2:account_id:key/key_id",
        "CreationDate": "time",
        "Enabled": true,
        "Description": "",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "KeyState": "Enabled",
        "Origin": "AWS_KMS",
        "KeyManager": "CUSTOMER",
        "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "KeySpec": "SYMMETRIC_DEFAULT",
        "EncryptionAlgorithms": [
            "SYMMETRIC_DEFAULT"
        ],
        "MultiRegion": false
    }
```

3. Utilizza un editor di testo per creare un file denominato `bucketencryption.json` con il seguente contenuto.

```
{
  "Rules": [
    {
      "ApplyServerSideEncryptionByDefault": {
        "SSEAlgorithm": "aws:kms",
        "KMSMasterKeyID": "{KMS Key ARN}"
      },
      "BucketKeyEnabled": true
    }
  ]
}
```

4. Inserisci il comando seguente, sostituendo *`bucket-name`* con il nome del bucket nel quale vuoi esportare i log.

```
aws s3api put-bucket-encryption --bucket bucket-name --server-side-encryption-
configuration file://bucketencryption.json
```

Se il comando non restituisce un errore, il processo ha esito positivo.

Passaggio 5: Creazione di un'attività di esportazione

Usare il comando seguente per creare l'attività di esportazione. Dopo averla creata, potrebbero essere necessari da pochi secondi ad alcune ore per l'attività di esportazione, in base alla dimensione dei dati da esportare.

Per esportare dati in Amazon S3 utilizzando il AWS CLI

1. Accedi con le autorizzazioni sufficienti come documentato in Fase 2: configurare le autorizzazioni di accesso.
2. Al prompt dei comandi, usa il seguente create-export-taskcomando per creare l'attività di esportazione.

```
aws logs create-export-task --profile CWLExportUser --task-name "my-log-
group-09-10-2015" --log-group-name "my-log-group" --from 1441490400000 --
to 1441494000000 --destination "my-exported-logs" --destination-prefix "export-
task-output"
```

Di seguito è riportato un output di esempio.

```
{
    "taskId": "cda45419-90ea-4db5-9833-aade86253e66"
}
```

Descrizione dei processi di esportazione

Dopo aver creato un'attività di esportazione, puoi ottenere lo stato corrente dell'attività.

Per descrivere le attività di esportazione utilizzando il AWS CLI

Al prompt dei comandi, utilizzare il describe-export-taskscomando seguente.

```
aws logs --profile CWLExportUser describe-export-tasks --task-id
 "cda45419-90ea-4db5-9833-aade86253e66"
```

Di seguito è riportato un output di esempio.

```
{
   "exportTasks": [
```

```
   {
      "destination": "my-exported-logs",
      "destinationPrefix": "export-task-output",
      "executionInfo": {
         "creationTime": 1441495400000
      },
      "from": 1441490400000,
      "logGroupName": "my-log-group",
      "status": {
         "code": "RUNNING",
         "message": "Started Successfully"
      },
      "taskId": "cda45419-90ea-4db5-9833-aade86253e66",
      "taskName": "my-log-group-09-10-2015",
      "tTo": 1441494000000
   }]
}
```

Puoi utilizzare il comando `describe-export-tasks` in tre diversi modi:

- Senza filtri: elenca tutte le attività di esportazione, in ordine inverso di creazione.
- Filtro su ID attività: elenca l'attività di esportazione, se esistente, con l'ID specificato.
- Filtro su stato dell'attività: elenca le attività di esportazione con lo stato specificato.

Ad esempio, utilizzare il seguente comando per applicare un filtro allo stato `FAILED`.

```
aws logs --profile CWLExportUser describe-export-tasks --status-code "FAILED"
```

Di seguito è riportato un output di esempio.

```
{
   "exportTasks": [
   {
      "destination": "my-exported-logs",
      "destinationPrefix": "export-task-output",
      "executionInfo": {
         "completionTime": 1441498600000
         "creationTime": 1441495400000
      },
      "from": 1441490400000,
      "logGroupName": "my-log-group",
```

```
      "status": {
         "code": "FAILED",
         "message": "FAILED"
      },
      "taskId": "cda45419-90ea-4db5-9833-aade86253e66",
      "taskName": "my-log-group-09-10-2015",
      "to": 1441494000000
   }]
}
```

Annullamento di un processo di esportazione

Puoi annullare un'attività di esportazione se il relativo stato è `PENDING` o `RUNNING`.

Per annullare un'operazione di esportazione utilizzando il AWS CLI

Al prompt dei comandi, utilizzate il seguente cancel-export-taskcomando:

```
aws logs --profile CWLExportUser cancel-export-task --task-id "cda45419-90ea-4db5-9833-aade86253e66"
```

È possibile utilizzare il describe-export-taskscomando per verificare che l'operazione sia stata annullata correttamente.

Streaming di dati CloudWatch Logs al servizio Amazon OpenSearch Service

Puoi configurare un gruppo di registro di CloudWatch Logs per lo streaming dei dati ricevuti nel cluster Amazon OpenSearch Service quasi in tempo reale attraverso una sottoscrizione di CloudWatch Logs. Per ulteriori informazioni, consulta Elaborazione in tempo reale dei dati di log con le sottoscrizioni.

A seconda della quantità di dati di log in fase di streaming, potrebbe essere necessario impostare un limite di esecuzioni simultanee a livello di funzione sulla funzione. Per ulteriori informazioni, consulta Lambda function scaling (Dimensionamento della funzione Lambda).

> **Note**
>
> Lo streaming di grandi quantità di dati da CloudWatch Logs al servizio OpenSearch potrebbe causare costi di utilizzo elevati. Ti consigliamo di creare un budget nella console AWS Billing and Cost Management. Per ulteriori informazioni, consulta Managing your costs with AWS Budgets (Gestione dei costi con i budget AWS).

Prerequisiti

Prima di iniziare, crea un dominio del servizio OpenSearch. Il dominio può avere accesso pubblico o accesso VPC, ma non puoi modificare il tipo di accesso dopo aver creato il dominio. In seguito, potrai riesaminare le impostazioni del dominio del servizio OpenSearch e modificare la configurazione del cluster in base alla quantità di dati elaborati dal cluster. Per istruzioni su come creare un dominio, consulta Creating OpenSearch Service domains (Creazione di domini OpenSearch Service).

Per ulteriori informazioni sul servizio OpenSearch, consulta la Guida per gli sviluppatori del servizio OpenSearch di Amazon.

Sottoscrizione di un gruppo di log al servizio OpenSearch

Per sottoscrivere un gruppo di log al servizio OpenSearch, puoi utilizzare la console CloudWatch.

Sottoscrizione di un gruppo di log al servizio OpenSearch

1. Aprire la console CloudWatch all'indirizzo https://console.aws.amazon.com/cloudwatch/.
2. Nel pannello di navigazione, seleziona Log groups (Gruppi di log).
3. Selezionare il nome del gruppo di registro.
4. Scegli Actions (Operazioni), Subscription filters (Filtri di sottoscrizione), Create Amazon OpenSearch Service subscription filter (Crea filtro di sottoscrizione del servizio OpenSearch di Amazon).
5. Scegli se eseguire lo streaming in un cluster di questo account o di un altro account.
   - Se hai scelto questo account, selezionare il dominio creato nella fase precedente.
   - Se hai scelto un altro account, fornire il dominio ARN e l'endpoint.
6. Per Lambda IAM Execution Role (Ruolo di esecuzione IAM Lambda), scegli il ruolo IAM che Lambda deve utilizzare durante l'esecuzione di chiamate a OpenSearch.

   Il ruolo IAM scelto deve soddisfare i seguenti requisiti:

   - Deve avere `lambda.amazonaws.com` nella relazione di trust.
   - Deve includere la policy seguente:

   ```
   {
       "Version": "2012-10-17",
       "Statement": [
           {
               "Action": [
                   "es:*"
               ],
               "Effect": "Allow",
               "Resource": "arn:aws:es:region:account-id:domain/target-domain-name/
   *"
           }
       ]
   }
   ```

   - Se il dominio del servizio OpenSearch di destinazione utilizza l'accesso VPC, al ruolo deve essere collegata la policy AWSLambdaVPCAccessExecutionRole. Questa policy gestita da Amazon concede a Lambda l'accesso al VPC del cliente, permettendo a Lambda di scrivere all'endpoint OpenSearch nel VPC.
7. Per Log format (Formato log) scegli un formato per il registro.

8. In Subscription filter pattern (Modello del filtro sottoscrizioni) digita i termini o i modelli da individuare nei tuoi registri eventi. Ciò assicura di inviare al cluster OpenSearch solo i dati di interesse. Per ulteriori informazioni, consulta Creazione di parametri da log eventi mediante filtri.
9. (Opzionale) In Select log data to test (Seleziona i dati di registro per il test), seleziona un flusso di registro, quindi scegli Test pattern (Modello di test) per verificare che il filtro di ricerca restituisca i risultati previsti.
10. Quindi scegli Start streaming (Avvia streaming).

Esempi di codice per i CloudWatch log che utilizzano gli SDK AWS

I seguenti esempi di codice mostrano come utilizzare CloudWatch Logs con un kit di sviluppo AWS software (SDK).

Le operazioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le operazioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati e negli esempi tra servizi.

Esempi cross-service: applicazioni di esempio che funzionano su più servizi Servizi AWS.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consulta. Utilizzo dei CloudWatch log con un SDK AWS Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Esempi di codice

- Azioni per i CloudWatch log che utilizzano gli SDK AWS
  - Associa una AWS KMS chiave a un gruppo di CloudWatch log Logs utilizzando un SDK AWS
  - Annullare un'attività di esportazione CloudWatch dei registri utilizzando un SDK AWS
  - Crea un gruppo di CloudWatch log Logs utilizzando un SDK AWS
  - Crea un flusso di log CloudWatch di Logs utilizzando un SDK AWS
  - Crea un filtro di sottoscrizione CloudWatch Logs utilizzando un SDK AWS
  - Crea un'attività di esportazione CloudWatch dei log utilizzando un SDK AWS
  - Eliminare un gruppo di log CloudWatch Logs utilizzando un SDK AWS
  - Eliminare un filtro di sottoscrizione CloudWatch Logs utilizzando un SDK AWS
  - Descrivi i filtri CloudWatch di abbonamento di Logs utilizzando un SDK AWS
  - Descrivi le attività CloudWatch di esportazione dei log utilizzando un SDK AWS
  - Descrivi i gruppi CloudWatch di log di Logs utilizzando un SDK AWS
- Esempi interservizi per i CloudWatch log che utilizzano gli SDK AWS
  - Utilizzo degli eventi pianificati per richiamare una funzione Lambda

Azioni per i CloudWatch log che utilizzano gli SDK AWS

I seguenti esempi di codice mostrano come eseguire singole azioni di CloudWatch Logs con gli SDK. AWS Questi estratti richiamano l'API CloudWatch Logs e sono estratti di codice di programmi più grandi che devono essere eseguiti nel contesto. Ogni esempio include un collegamento a GitHub, dove è possibile trovare le istruzioni per la configurazione e l'esecuzione del codice.

Gli esempi seguenti includono solo le operazioni più comunemente utilizzate. Per un elenco completo, consulta l'Amazon CloudWatch Logs API Reference.

Esempi

- Associa una AWS KMS chiave a un gruppo di CloudWatch log Logs utilizzando un SDK AWS
- Annullare un'attività di esportazione CloudWatch dei registri utilizzando un SDK AWS
- Crea un gruppo di CloudWatch log Logs utilizzando un SDK AWS
- Crea un flusso di log CloudWatch di Logs utilizzando un SDK AWS
- Crea un filtro di sottoscrizione CloudWatch Logs utilizzando un SDK AWS
- Crea un'attività di esportazione CloudWatch dei log utilizzando un SDK AWS
- Eliminare un gruppo di log CloudWatch Logs utilizzando un SDK AWS
- Eliminare un filtro di sottoscrizione CloudWatch Logs utilizzando un SDK AWS
- Descrivi i filtri CloudWatch di abbonamento di Logs utilizzando un SDK AWS
- Descrivi le attività CloudWatch di esportazione dei log utilizzando un SDK AWS
- Descrivi i gruppi CloudWatch di log di Logs utilizzando un SDK AWS

Associa una AWS KMS chiave a un gruppo di CloudWatch log Logs utilizzando un SDK AWS

Il seguente esempio di codice mostra come associare una AWS KMS chiave a un gruppo di CloudWatch log Logs esistente.

.NET

AWS SDK for .NET

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codiceAWS.

```
    using System;
    using System.Threading.Tasks;
    using Amazon.CloudWatchLogs;
    using Amazon.CloudWatchLogs.Model;

    /// <summary>
    /// Shows how to associate an AWS Key Management Service (AWS KMS) key with
    /// an Amazon CloudWatch Logs log group. The example was created using the
    /// AWS SDK for .NET version 3.7 and .NET Core 5.0.
    /// </summary>
    public class AssociateKmsKey
    {
        public static async Task Main()
        {
            // This client object will be associated with the same AWS Region
            // as the default user on this system. If you need to use a
            // different AWS Region, pass it as a parameter to the client
            // constructor.
            var client = new AmazonCloudWatchLogsClient();

            string kmsKeyId = "arn:aws:kms:us-west-2:<account-
number>:key/7c9eccc2-38cb-4c4f-9db3-766ee8dd3ad4";
            string groupName = "cloudwatchlogs-example-loggroup";

            var request = new AssociateKmsKeyRequest
            {
                KmsKeyId = kmsKeyId,
                LogGroupName = groupName,
            };

            var response = await client.AssociateKmsKeyAsync(request);
```

```
            if (response.HttpStatusCode == System.Net.HttpStatusCode.OK)
            {
                Console.WriteLine($"Successfully associated KMS key ID:
{kmsKeyId} with log group: {groupName}.");
            }
            else
            {
                Console.WriteLine("Could not make the association between:
{kmsKeyId} and {groupName}.");
            }
        }
    }
```

- Per i dettagli sull'API, consulta la AssociateKmsKeysezione AWS SDK for .NET API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consultaUtilizzo dei CloudWatch log con un SDK AWS. Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Annullare un'attività di esportazione CloudWatch dei registri utilizzando un SDK AWS

Il seguente esempio di codice mostra come annullare un'attività di esportazione dei CloudWatch registri esistente.

.NET

AWS SDK for .NET

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codiceAWS.

```
    using System;
```

```
using System.Threading.Tasks;
using Amazon.CloudWatchLogs;
using Amazon.CloudWatchLogs.Model;

/// <summary>
/// Shows how to cancel an Amazon CloudWatch Logs export task. The example
/// uses the AWS SDK for .NET version 3.7 and .NET Core 5.0.
/// </summary>
public class CancelExportTask
{
    public static async Task Main()
    {
        // This client object will be associated with the same AWS Region
        // as the default user on this system. If you need to use a
        // different AWS Region, pass it as a parameter to the client
        // constructor.
        var client = new AmazonCloudWatchLogsClient();
        string taskId = "exampleTaskId";

        var request = new CancelExportTaskRequest
        {
            TaskId = taskId,
        };

        var response = await client.CancelExportTaskAsync(request);

        if (response.HttpStatusCode == System.Net.HttpStatusCode.OK)
        {
            Console.WriteLine($"{taskId} successfully canceled.");
        }
        else
        {
            Console.WriteLine($"{taskId} could not be canceled.");
        }
    }
}
```

- Per i dettagli sull'API, consulta la CancelExportTasksezione AWS SDK for .NET API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consultaUtilizzo dei CloudWatch log con un SDK AWS. Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Crea un gruppo di CloudWatch log Logs utilizzando un SDK AWS

I seguenti esempi di codice mostrano come creare un nuovo gruppo di CloudWatch log Logs.

.NET

AWS SDK for .NET

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codiceAWS.

```
using System;
using System.Threading.Tasks;
using Amazon.CloudWatchLogs;
using Amazon.CloudWatchLogs.Model;

/// <summary>
/// Shows how to create an Amazon CloudWatch Logs log group. The example
/// was created using the AWS SDK for .NET version 3.7 and .NET Core 5.0.
/// </summary>
public class CreateLogGroup
{
    public static async Task Main()
    {
        // This client object will be associated with the same AWS Region
        // as the default user on this system. If you need to use a
        // different AWS Region, pass it as a parameter to the client
        // constructor.
        var client = new AmazonCloudWatchLogsClient();

        string logGroupName = "cloudwatchlogs-example-loggroup";

        var request = new CreateLogGroupRequest
        {
```

```
                LogGroupName = logGroupName,
            };

            var response = await client.CreateLogGroupAsync(request);

            if (response.HttpStatusCode == System.Net.HttpStatusCode.OK)
            {
                Console.WriteLine($"Successfully create log group with ID:
{logGroupName}.");
            }
            else
            {
                Console.WriteLine("Could not create log group.");
            }
        }
    }
```

- Per i dettagli sull'API, consulta la CreateLogGroupsezione AWS SDK for .NET API Reference.

JavaScript

SDK per JavaScript (v3)

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codiceAWS.

```
import { CreateLogGroupCommand } from "@aws-sdk/client-cloudwatch-logs";
import { client } from "../libs/client.js";

const run = async () => {
  const command = new CreateLogGroupCommand({
    // The name of the log group.
    logGroupName: process.env.CLOUDWATCH_LOGS_LOG_GROUP,
  });
```

```
  try {
    return await client.send(command);
  } catch (err) {
    console.error(err);
  }
};

export default run();
```

- Per i dettagli sull'API, consulta la CreateLogGroupsezione AWS SDK for JavaScript API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consultaUtilizzo dei CloudWatch log con un SDK AWS. Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Crea un flusso di log CloudWatch di Logs utilizzando un SDK AWS

Il seguente esempio di codice mostra come creare un nuovo flusso di log di CloudWatch Logs.

.NET

AWS SDK for .NET

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codiceAWS.

```
using System;
using System.Threading.Tasks;
using Amazon.CloudWatchLogs;
using Amazon.CloudWatchLogs.Model;

/// <summary>
/// Shows how to create an Amazon CloudWatch Logs stream for a CloudWatch
/// log group. The example was created using the AWS SDK for .NET version
/// 3.7 and .NET Core 5.0.
```

```
    /// </summary>
    public class CreateLogStream
    {
        public static async Task Main()
        {
            // This client object will be associated with the same AWS Region
            // as the default user on this system. If you need to use a
            // different AWS Region, pass it as a parameter to the client
            // constructor.
            var client = new AmazonCloudWatchLogsClient();
            string logGroupName = "cloudwatchlogs-example-loggroup";
            string logStreamName = "cloudwatchlogs-example-logstream";

            var request = new CreateLogStreamRequest
            {
                LogGroupName = logGroupName,
                LogStreamName = logStreamName,
            };

            var response = await client.CreateLogStreamAsync(request);

            if (response.HttpStatusCode == System.Net.HttpStatusCode.OK)
            {
                Console.WriteLine($"{logStreamName} successfully created for
{logGroupName}.");
            }
            else
            {
                Console.WriteLine("Could not create stream.");
            }
        }
    }
```

- Per i dettagli sull'API, consulta la CreateLogStreamsezione AWS SDK for .NET API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consultaUtilizzo dei CloudWatch log con un SDK AWS. Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Crea un filtro di sottoscrizione CloudWatch Logs utilizzando un SDK AWS

I seguenti esempi di codice mostrano come creare un filtro di abbonamento Amazon CloudWatch Logs.

C++

SDK per C++

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri come configurarlo ed eseguirlo nel AWS Code Examples Repository.

Includere i file richiesti.

```
#include <aws/core/Aws.h>
#include <aws/logs/CloudWatchLogsClient.h>
#include <aws/logs/model/PutSubscriptionFilterRequest.h>
#include <aws/core/utils/Outcome.h>
#include <iostream>
```

Creare il filtro di sottoscrizione.

```
        Aws::CloudWatchLogs::CloudWatchLogsClient cwl;
        Aws::CloudWatchLogs::Model::PutSubscriptionFilterRequest request;
        request.SetFilterName(filter_name);
        request.SetFilterPattern(filter_pattern);
        request.SetLogGroupName(log_group);
        request.SetDestinationArn(dest_arn);
        auto outcome = cwl.PutSubscriptionFilter(request);
        if (!outcome.IsSuccess())
        {
            std::cout << "Failed to create CloudWatch logs subscription filter "
                << filter_name << ": " << outcome.GetError().GetMessage() <<
                std::endl;
        }
        else
        {
```

```
            std::cout << "Successfully created CloudWatch logs subscription " <<
                "filter " << filter_name << std::endl;
        }
```

- Per i dettagli sull'API, consulta la PutSubscriptionFiltersezione AWS SDK for C++ API Reference.

Java

SDK per Java 2.x

> **Note**
>
> C'è altro su GitHub. Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codiceAWS.

```
   public static void putSubFilters(CloudWatchLogsClient cwl,
                                    String filter,
                                    String pattern,
                                    String logGroup,
                                    String functionArn) {

       try {
           PutSubscriptionFilterRequest request =
PutSubscriptionFilterRequest.builder()
                .filterName(filter)
                .filterPattern(pattern)
                .logGroupName(logGroup)
                .destinationArn(functionArn)
                .build();

           cwl.putSubscriptionFilter(request);
           System.out.printf(
                   "Successfully created CloudWatch logs subscription filter
%s",
                   filter);

       } catch (CloudWatchLogsException e) {
           System.err.println(e.awsErrorDetails().errorMessage());
```

```
            System.exit(1);
        }
    }
```

- Per i dettagli sull'API, consulta la PutSubscriptionFiltersezione AWS SDK for Java 2.x API Reference.

JavaScript

SDK per JavaScript (v3)

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codiceAWS.

```
import { PutSubscriptionFilterCommand } from "@aws-sdk/client-cloudwatch-logs";
import { client } from "../libs/client.js";

const run = async () => {
  const command = new PutSubscriptionFilterCommand({
    // An ARN of a same-account Kinesis stream, Kinesis Firehose
    // delivery stream, or Lambda function.
    // https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/
SubscriptionFilters.html
    destinationArn: process.env.CLOUDWATCH_LOGS_DESTINATION_ARN,

    // A name for the filter.
    filterName: process.env.CLOUDWATCH_LOGS_FILTER_NAME,

    // A filter pattern for subscribing to a filtered stream of log events.
    // https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/
FilterAndPatternSyntax.html
    filterPattern: process.env.CLOUDWATCH_LOGS_FILTER_PATTERN,

    // The name of the log group. Messages in this group matching the filter
 pattern
    // will be sent to the destination ARN.
    logGroupName: process.env.CLOUDWATCH_LOGS_LOG_GROUP,
```

```
  });

  try {
    return await client.send(command);
  } catch (err) {
    console.error(err);
  }
};

export default run();
```

- Per i dettagli sull'API, consulta la PutSubscriptionFiltersezione AWS SDK for JavaScript API Reference.

SDK per JavaScript (v2)

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codiceAWS.

```
// Load the AWS SDK for Node.js
var AWS = require('aws-sdk');
// Set the region
AWS.config.update({region: 'REGION'});

// Create the CloudWatchLogs service object
var cwl = new AWS.CloudWatchLogs({apiVersion: '2014-03-28'});

var params = {
  destinationArn: 'LAMBDA_FUNCTION_ARN',
  filterName: 'FILTER_NAME',
  filterPattern: 'ERROR',
  logGroupName: 'LOG_GROUP',
};

cwl.putSubscriptionFilter(params, function(err, data) {
  if (err) {
    console.log("Error", err);
  } else {
```

```
    console.log("Success", data);
  }
});
```

- Per ulteriori informazioni, consulta la Guida per sviluppatori diAWS SDK for JavaScript.
- Per i dettagli sull'API, consulta la PutSubscriptionFiltersezione AWS SDK for JavaScript API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consultaUtilizzo dei CloudWatch log con un SDK AWS. Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Crea un'attività di esportazione CloudWatch dei log utilizzando un SDK AWS

Il seguente esempio di codice mostra come creare una nuova attività di esportazione CloudWatch dei registri.

.NET

AWS SDK for .NET

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codiceAWS.

```
    using System;
    using System.Threading.Tasks;
    using Amazon.CloudWatchLogs;
    using Amazon.CloudWatchLogs.Model;

    /// <summary>
    /// Shows how to create an Export Task to export the contents of the Amazon
    /// CloudWatch Logs to the specified Amazon Simple Storage Service (Amazon
S3)
```

```
    /// bucket. The example was created with the AWS SDK for .NET version 3.7 and
    /// .NET Core 5.0.
    /// </summary>
    public class CreateExportTask
    {
        public static async Task Main()
        {
            // This client object will be associated with the same AWS Region
            // as the default user on this system. If you need to use a
            // different AWS Region, pass it as a parameter to the client
            // constructor.
            var client = new AmazonCloudWatchLogsClient();
            string taskName = "export-task-example";
            string logGroupName = "cloudwatchlogs-example-loggroup";
            string destination = "doc-example-bucket";
            var fromTime = 1437584472382;
            var toTime = 1437584472833;

            var request = new CreateExportTaskRequest
            {
                From = fromTime,
                To = toTime,
                TaskName = taskName,
                LogGroupName = logGroupName,
                Destination = destination,
            };

            var response = await client.CreateExportTaskAsync(request);

            if (response.HttpStatusCode == System.Net.HttpStatusCode.OK)
            {
                Console.WriteLine($"The task, {taskName} with ID: " +
                                  $"{response.TaskId} has been created
successfully.");
            }
        }
    }
```

- Per i dettagli sull'API, consulta la CreateExportTasksezione AWS SDK for .NET API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consultaUtilizzo dei CloudWatch log con un SDK AWS. Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Eliminare un gruppo di log CloudWatch Logs utilizzando un SDK AWS

I seguenti esempi di codice mostrano come eliminare un gruppo di CloudWatch log Logs esistente.

.NET

AWS SDK for .NET

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codiceAWS.

```
using System;
using System.Threading.Tasks;
using Amazon.CloudWatchLogs;
using Amazon.CloudWatchLogs.Model;

/// <summary>
/// Uses the Amazon CloudWatch Logs Service to delete an existing
/// CloudWatch Logs log group. The example was created using the
/// AWS SDK for .NET version 3.7 and .NET Core 5.0.
/// </summary>
public class DeleteLogGroup
{
    public static async Task Main()
    {
        var client = new AmazonCloudWatchLogsClient();
        string logGroupName = "cloudwatchlogs-example-loggroup";

        var request = new DeleteLogGroupRequest
        {
            LogGroupName = logGroupName,
        };

        var response = await client.DeleteLogGroupAsync(request);
```

```
            if (response.HttpStatusCode == System.Net.HttpStatusCode.OK)
            {
                Console.WriteLine($"Successfully deleted CloudWatch log group,
 {logGroupName}.");
            }
        }
    }
```

- Per i dettagli sull'API, consulta la DeleteLogGroupsezione AWS SDK for .NET API Reference.

JavaScript

SDK per JavaScript (v3)

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codiceAWS.

```
import { DeleteLogGroupCommand } from "@aws-sdk/client-cloudwatch-logs";
import { client } from "../libs/client.js";

const run = async () => {
  const command = new DeleteLogGroupCommand({
    // The name of the log group.
    logGroupName: process.env.CLOUDWATCH_LOGS_LOG_GROUP,
  });

  try {
    return await client.send(command);
  } catch (err) {
    console.error(err);
  }
};

export default run();
```

- Per i dettagli sull'API, consulta la DeleteLogGroupsezione AWS SDK for JavaScript API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consultaUtilizzo dei CloudWatch log con un SDK AWS. Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Eliminare un filtro di sottoscrizione CloudWatch Logs utilizzando un SDK AWS

I seguenti esempi di codice mostrano come eliminare un filtro di abbonamento Amazon CloudWatch Logs.

C++

SDK per C++

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri come configurarlo ed eseguirlo nel AWS Code Examples Repository.

Includere i file richiesti.

```
#include <aws/core/Aws.h>
#include <aws/core/utils/Outcome.h>
#include <aws/logs/CloudWatchLogsClient.h>
#include <aws/logs/model/DeleteSubscriptionFilterRequest.h>
#include <iostream>
```

Eliminare il filtro di sottoscrizione.

```
        Aws::CloudWatchLogs::CloudWatchLogsClient cwl;
        Aws::CloudWatchLogs::Model::DeleteSubscriptionFilterRequest request;
```

```
        request.SetFilterName(filter_name);
        request.SetLogGroupName(log_group);

        auto outcome = cwl.DeleteSubscriptionFilter(request);
        if (!outcome.IsSuccess()) {
            std::cout << "Failed to delete CloudWatch log subscription filter "
                << filter_name << ": " << outcome.GetError().GetMessage() <<
                std::endl;
        } else {
            std::cout << "Successfully deleted CloudWatch logs subscription " <<
                "filter " << filter_name << std::endl;
        }
```

- Per i dettagli sull'API, consulta la DeleteSubscriptionFiltersezione AWS SDK for C++ API Reference.

Java

SDK per Java 2.x

> **Note**
>
> C'è altro su GitHub. Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codiceAWS.

```
   public static void deleteSubFilter(CloudWatchLogsClient logs, String filter,
String logGroup) {

        try {
            DeleteSubscriptionFilterRequest request =
DeleteSubscriptionFilterRequest.builder()
                .filterName(filter)
                .logGroupName(logGroup)
                .build();

            logs.deleteSubscriptionFilter(request);
            System.out.printf("Successfully deleted CloudWatch logs subscription
filter %s", filter);
```

```
        } catch (CloudWatchException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }
```

- Per i dettagli sull'API, consulta la DeleteSubscriptionFiltersezione AWS SDK for Java 2.x API Reference.

JavaScript

SDK per JavaScript (v3)

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codiceAWS.

```
import { DeleteSubscriptionFilterCommand } from "@aws-sdk/client-cloudwatch-
logs";
import { client } from "../libs/client.js";

const run = async () => {
  const command = new DeleteSubscriptionFilterCommand({
    // The name of the filter.
    filterName: process.env.CLOUDWATCH_LOGS_FILTER_NAME,
    // The name of the log group.
    logGroupName: process.env.CLOUDWATCH_LOGS_LOG_GROUP,
  });

  try {
    return await client.send(command);
  } catch (err) {
    console.error(err);
  }
};

export default run();
```

- Per i dettagli sull'API, consulta la DeleteSubscriptionFiltersezione AWS SDK for JavaScript API Reference.

SDK per JavaScript (v2)

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codiceAWS.

```
// Load the AWS SDK for Node.js
var AWS = require('aws-sdk');
// Set the region
AWS.config.update({region: 'REGION'});

// Create the CloudWatchLogs service object
var cwl = new AWS.CloudWatchLogs({apiVersion: '2014-03-28'});

var params = {
  filterName: 'FILTER',
  logGroupName: 'LOG_GROUP'
};

cwl.deleteSubscriptionFilter(params, function(err, data) {
  if (err) {
    console.log("Error", err);
  } else {
    console.log("Success", data);
  }
});
```

- Per ulteriori informazioni, consulta la Guida per sviluppatori diAWS SDK for JavaScript.
- Per i dettagli sull'API, consulta la DeleteSubscriptionFiltersezione AWS SDK for JavaScript API Reference.

Kotlin

SDK per Kotlin

> **Note**
>
> Si tratta di una documentazione di pre-rilascio di una caratteristica nella versione di anteprima. ed è soggetta a modifiche.

> **Note**
>
> C'è altro su GitHub. Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codiceAWS.

```
suspend fun deleteSubFilter(filter: String?, logGroup: String?) {

    val request = DeleteSubscriptionFilterRequest {
        filterName = filter
        logGroupName = logGroup
    }

    CloudWatchLogsClient { region = "us-west-2" }.use { logs ->
        logs.deleteSubscriptionFilter(request)
        println("Successfully deleted CloudWatch logs subscription filter named
 $filter")
    }
}
```

- Per i dettagli sull'API, DeleteSubscriptionFilterconsulta AWS SDK for Kotlin API reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consulta. Utilizzo dei CloudWatch log con un SDK AWS Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Descrivi i filtri CloudWatch di abbonamento di Logs utilizzando un SDK AWS

I seguenti esempi di codice mostrano come descrivere i filtri di abbonamento esistenti di Amazon CloudWatch Logs.

C++

SDK per C++

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri come configurarlo ed eseguirlo nel AWS Code Examples Repository.

Includere i file richiesti.

```
#include <aws/core/Aws.h>
#include <aws/core/utils/Outcome.h>
#include <aws/logs/CloudWatchLogsClient.h>
#include <aws/logs/model/DescribeSubscriptionFiltersRequest.h>
#include <aws/logs/model/DescribeSubscriptionFiltersResult.h>
#include <iostream>
#include <iomanip>
```

Elencare i filtri di sottoscrizione.

```
        Aws::CloudWatchLogs::CloudWatchLogsClient cwl;
        Aws::CloudWatchLogs::Model::DescribeSubscriptionFiltersRequest request;
        request.SetLogGroupName(log_group);
        request.SetLimit(1);

        bool done = false;
        bool header = false;
        while (!done) {
            auto outcome = cwl.DescribeSubscriptionFilters(
                    request);
            if (!outcome.IsSuccess()) {
```

```
                std::cout << "Failed to describe CloudWatch subscription filters
"
                    << "for log group " << log_group << ": " <<
                    outcome.GetError().GetMessage() << std::endl;
                break;
            }

            if (!header) {
                std::cout << std::left << std::setw(32) << "Name" <<
                    std::setw(64) << "FilterPattern" << std::setw(64) <<
                    "DestinationArn" << std::endl;
                header = true;
            }

            const auto &filters = outcome.GetResult().GetSubscriptionFilters();
            for (const auto &filter : filters) {
                std::cout << std::left << std::setw(32) <<
                    filter.GetFilterName() << std::setw(64) <<
                    filter.GetFilterPattern() << std::setw(64) <<
                    filter.GetDestinationArn() << std::endl;
            }

            const auto &next_token = outcome.GetResult().GetNextToken();
            request.SetNextToken(next_token);
            done = next_token.empty();
        }
```

- Per i dettagli sull'API, consulta la DescribeSubscriptionFilterssezione AWS SDK for C++ API Reference.

Java

SDK per Java 2.x

> **Note**
>
> C'è altro su GitHub. Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codiceAWS.

```
   public static void describeFilters(CloudWatchLogsClient logs, String
logGroup) {

       try {
           boolean done = false;
           String newToken = null;

           while(!done) {
               DescribeSubscriptionFiltersResponse response;
               if (newToken == null) {
                   DescribeSubscriptionFiltersRequest request =
DescribeSubscriptionFiltersRequest.builder()
                           .logGroupName(logGroup)
                           .limit(1).build();

                   response = logs.describeSubscriptionFilters(request);
               } else {
                   DescribeSubscriptionFiltersRequest request =
DescribeSubscriptionFiltersRequest.builder()
                           .nextToken(newToken)
                           .logGroupName(logGroup)
                           .limit(1).build();
                   response = logs.describeSubscriptionFilters(request);
               }

               for(SubscriptionFilter filter : response.subscriptionFilters()) {
                   System.out.printf("Retrieved filter with name %s, " +
"pattern %s " + "and destination arn %s",
                           filter.filterName(),
                           filter.filterPattern(),
                           filter.destinationArn());
               }

               if(response.nextToken() == null) {
                   done = true;
               } else {
                   newToken = response.nextToken();
               }
           }

       } catch (CloudWatchException e) {
           System.err.println(e.awsErrorDetails().errorMessage());
           System.exit(1);
```

```
        }
        System.out.printf("Done");
    }
```

- Per i dettagli sull'API, consulta la DescribeSubscriptionFilterssezione AWS SDK for Java 2.x API Reference.

JavaScript

SDK per JavaScript (v3)

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codiceAWS.

```
import { DescribeSubscriptionFiltersCommand } from "@aws-sdk/client-cloudwatch-
logs";
import { client } from "../libs/client.js";

const run = async () => {
  // This will return a list of all subscription filters in your account
  // matching the log group name.
  const command = new DescribeSubscriptionFiltersCommand({
    logGroupName: process.env.CLOUDWATCH_LOGS_LOG_GROUP,
    limit: 1,
  });

  try {
    return await client.send(command);
  } catch (err) {
    console.error(err);
  }
};

export default run();
```

- Per i dettagli sull'API, consulta la DescribeSubscriptionFilterssezione AWS SDK for JavaScript API Reference.

SDK per JavaScript (v2)

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codiceAWS.

```
// Load the AWS SDK for Node.js
var AWS = require('aws-sdk');
// Set the region
AWS.config.update({region: 'REGION'});

// Create the CloudWatchLogs service object
var cwl = new AWS.CloudWatchLogs({apiVersion: '2014-03-28'});

var params = {
  logGroupName: 'GROUP_NAME',
  limit: 5
};

cwl.describeSubscriptionFilters(params, function(err, data) {
  if (err) {
    console.log("Error", err);
  } else {
    console.log("Success", data.subscriptionFilters);
  }
});
```

- Per ulteriori informazioni, consulta la Guida per sviluppatori diAWS SDK for JavaScript.
- Per i dettagli sull'API, consulta la DescribeSubscriptionFilterssezione AWS SDK for JavaScript API Reference.

Kotlin

SDK per Kotlin

> **Note**
>
> Si tratta di una documentazione di pre-rilascio di una caratteristica nella versione di anteprima. ed è soggetta a modifiche.

> **Note**
>
> C'è altro su GitHub. Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codiceAWS.

```
suspend fun describeFilters(logGroup: String) {

    val request = DescribeSubscriptionFiltersRequest {
        logGroupName = logGroup
        limit = 1
    }

    CloudWatchLogsClient { region = "us-west-2" }.use { cwlClient ->
        val response = cwlClient.describeSubscriptionFilters(request)
        response.subscriptionFilters?.forEach { filter ->
            println("Retrieved filter with name  ${filter.filterName} pattern
 ${filter.filterPattern} and destination ${filter.destinationArn}")
        }
    }
}
```

- Per i dettagli sull'API, DescribeSubscriptionFiltersconsulta AWS SDK for Kotlin API reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consulta. Utilizzo dei CloudWatch log con un SDK AWS Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Descrivi le attività CloudWatch di esportazione dei log utilizzando un SDK AWS

Il seguente esempio di codice mostra come descrivere le attività di esportazione CloudWatch dei log.

.NET

AWS SDK for .NET

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codiceAWS.

```
using System;
using System.Threading.Tasks;
using Amazon.CloudWatchLogs;
using Amazon.CloudWatchLogs.Model;

/// <summary>
/// Shows how to retrieve a list of information about Amazon CloudWatch
/// Logs export tasks. The example was created using the AWS SDK for .NET
/// version 3.7 and .NET Core 5.0.
/// </summary>
public class DescribeExportTasks
{
    public static async Task Main()
    {
        // This client object will be associated with the same AWS Region
        // as the default user on this system. If you need to use a
        // different AWS Region, pass it as a parameter to the client
        // constructor.
        var client = new AmazonCloudWatchLogsClient();

        var request = new DescribeExportTasksRequest
        {
            Limit = 5,
        };

        var response = new DescribeExportTasksResponse();
```

```
            do
            {
                response = await client.DescribeExportTasksAsync(request);
                response.ExportTasks.ForEach(t =>
                {
                    Console.WriteLine($"{t.TaskName} with ID: {t.TaskId} has
status: {t.Status}");
                });
            }
            while (response.NextToken is not null);
        }
    }
```

- Per i dettagli sull'API, consulta la DescribeExportTaskssezione AWS SDK for .NET API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consultaUtilizzo dei CloudWatch log con un SDK AWS. Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Descrivi i gruppi CloudWatch di log di Logs utilizzando un SDK AWS

I seguenti esempi di codice mostrano come descrivere i gruppi di CloudWatch log Logs.

.NET

AWS SDK for .NET

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codiceAWS.

```
using System;
using System.Threading.Tasks;
using Amazon.CloudWatchLogs;
```

```
    using Amazon.CloudWatchLogs.Model;

    /// <summary>
    /// Retrieves information about existing Amazon CloudWatch Logs log groups
    /// and displays the information on the console. The example was created
    /// using the AWS SDK for .NET version 3.7 and .NET Core 5.0.
    /// </summary>
    public class DescribeLogGroups
    {
        public static async Task Main()
        {
            // Creates a CloudWatch Logs client using the default
            // user. If you need to work with resources in another
            // AWS Region than the one defined for the default user,
            // pass the AWS Region as a parameter to the client constructor.
            var client = new AmazonCloudWatchLogsClient();

            bool done = false;
            string? newToken = null;

            var request = new DescribeLogGroupsRequest
            {
                Limit = 5,
            };

            DescribeLogGroupsResponse response;

            do
            {
                if (newToken is not null)
                {
                    request.NextToken = newToken;
                }

                response = await client.DescribeLogGroupsAsync(request);

                response.LogGroups.ForEach(lg =>
                {
                    Console.WriteLine($"{lg.LogGroupName} is associated with the
key: {lg.KmsKeyId}.");
                    Console.WriteLine($"Created on:
{lg.CreationTime.Date.Date}");
                    Console.WriteLine($"Date for this group will be stored for:
{lg.RetentionInDays} days.\n");
```

```
                });

                if (response.NextToken is null)
                {
                    done = true;
                }
                else
                {
                    newToken = response.NextToken;
                }
            }
            while (!done);
        }
    }
```

- Per i dettagli sull'API, consulta la DescribeLogGroupssezione AWS SDK for .NET API Reference.

JavaScript

SDK per JavaScript (v3)

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codiceAWS.

```
import {
  paginateDescribeLogGroups,
  CloudWatchLogsClient,
} from "@aws-sdk/client-cloudwatch-logs";

const client = new CloudWatchLogsClient({});

export const main = async () => {
  const paginatedLogGroups = paginateDescribeLogGroups({ client }, {});
  const logGroups = [];
```

```
  for await (const page of paginatedLogGroups) {
    if (page.logGroups && page.logGroups.every((lg) => !!lg)) {
      logGroups.push(...page.logGroups);
    }
  }

  console.log(logGroups);
  return logGroups;
};
```

- Per i dettagli sull'API, consulta la DescribeLogGroupssezione AWS SDK for JavaScript API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consultaUtilizzo dei CloudWatch log con un SDK AWS. Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Esempi interservizi per i CloudWatch log che utilizzano gli SDK AWS

Le seguenti applicazioni di esempio utilizzano AWS gli SDK per combinare i CloudWatch log con altri. Servizi AWSOgni esempio include un collegamento a GitHub, dove è possibile trovare istruzioni su come configurare ed eseguire l'applicazione.

Esempi

- Utilizzo degli eventi pianificati per richiamare una funzione Lambda

Utilizzo degli eventi pianificati per richiamare una funzione Lambda

I seguenti esempi di codice mostrano come creare una AWS Lambda funzione richiamata da un evento EventBridge pianificato di Amazon.

Python

SDK per Python (Boto3)

Questo esempio mostra come registrare una AWS Lambda funzione come destinazione di un EventBridge evento Amazon pianificato. Il gestore Lambda scrive un messaggio intuitivo

e i dati completi dell'evento su Amazon CloudWatch Logs per recuperarli in un secondo momento.

- Distribuzione di una funzione Lambda.
- Crea un evento EventBridge pianificato e rende la funzione Lambda la destinazione.
- Concede il permesso di EventBridge invocare la funzione Lambda.
- Stampa i dati più recenti dai CloudWatch registri per mostrare il risultato delle chiamate pianificate.
- Elimina tutte le risorse create durante la demo.

Questo esempio è visualizzato al meglio su. GitHub Per il codice sorgente completo e le istruzioni su come configurarlo ed eseguirlo, vedi l'esempio completo su GitHub.

Servizi utilizzati in questo esempio

- CloudWatch Registri
- EventBridge
- Lambda

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consulta. Utilizzo dei CloudWatch log con un SDK AWS Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Sicurezza in Amazon CloudWatch Logs

Per AWS, la sicurezza del cloud ha la massima priorità. In quanto cliente AWS, è possibile trarre vantaggio da un'architettura di data center e di rete progettata per soddisfare i requisiti delle organizzazioni più esigenti a livello di sicurezza.

La sicurezza è una responsabilità condivisa tra te e AWS. Il modello di responsabilità condivisa descrive questo modello come sicurezza del cloud e sicurezza nel cloud:

- Sicurezza del cloud - AWS è responsabile della protezione dell'infrastruttura che esegue i servizi AWS in AWS Cloud. AWS fornisce inoltre servizi che è possibile utilizzare in modo sicuro. I revisori di terze parti testano regolarmente e verificano l'efficacia della nostra sicurezza nell'ambito dei Programmi di conformità AWS. Per informazioni sui programmi di conformità applicabili a WorkSpaces, consulta AWS Services in Scope by Compliance Program (Servizi coperti dal programma di conformità).
- Sicurezza nel cloud: la tua responsabilità è determinata dal servizio AWS che utilizzi. Sei anche responsabile di altri fattori, tra cui la riservatezza dei dati, i tuoi requisiti aziendali e le leggi e le normative applicabili

Questa documentazione aiuta a comprendere come applicare il modello di responsabilità condivisa quando si usa Amazon CloudWatch Logs. Viene illustrato come configurare Amazon CloudWatch Logs per soddisfare gli obiettivi di sicurezza e conformità. È inoltre illustrato come utilizzare altri servizi AWS che consentono di monitorare e proteggere le risorse CloudWatch Logs.

Indice

Protezione dei dati in Amazon CloudWatch Logs

> **Note**
>
> Oltre alle seguenti informazioni sulla protezione generale dei dati in AWS, CloudWatch Logs consente anche di proteggere i dati sensibili nei log eventi attraverso il mascheramento. Per ulteriori informazioni, consulta Incremento della protezione dei dati di log sensibili con il mascheramento.

Il modello di responsabilità condivisa di AWS si applica alla protezione dei dati in Amazon CloudWatch Logs. Come descritto in questo modello, AWS è responsabile della protezione dell'infrastruttura globale che esegue tutto l'Cloud AWS. L'utente è responsabile di mantenere il controllo sui contenuti ospitati su questa infrastruttura. Questi contenuti comprendono la configurazione della protezione e le attività di gestione per i Servizi AWS utilizzati. Per ulteriori informazioni sulla privacy dei dati, vedi le Domande frequenti sulla privacy dei dati. Per informazioni sulla protezione dei dati in Europa, consulta il post del blog relativo al Modello di responsabilità condivisa AWS e GDPR nel Blog sulla sicurezza AWS.

Per garantire la protezione dei dati, ti suggeriamo di proteggere le credenziali Account AWS e di configurare i singoli utenti con AWS IAM Identity Center o AWS Identity and Access Management (IAM). In questo modo, a ogni utente verranno assegnate solo le autorizzazioni necessarie per svolgere il suo lavoro. Ti suggeriamo, inoltre, di proteggere i dati nei seguenti modi:

- Utilizza l'autenticazione a più fattori (MFA) con ogni account.
- Utilizza SSL/TLS per comunicare con le risorse AWS. È richiesto TLS 1.2 e consigliato TLS 1.3.
- Configura la registrazione delle API e delle attività degli utenti con AWS CloudTrail.
- Utilizza le soluzioni di crittografia AWS, insieme a tutti i controlli di sicurezza di default all'interno dei Servizi AWS.
- Utilizza i servizi di sicurezza gestiti avanzati, ad esempio Amazon Macie, che aiutano a individuare e proteggere i dati sensibili archiviati in Amazon S3.
- Se si richiedono moduli crittografici convalidati FIPS 140-2 quando si accede ad AWS tramite una CLI o un'API, utilizzare un endpoint FIPS. Per ulteriori informazioni sugli endpoint FIPS disponibili, consulta il Federal Information Processing Standard (FIPS) 140-2.

Ti consigliamo di non inserire mai informazioni riservate o sensibili, ad esempio, gli indirizzi e-mail dei clienti, nei tag o nei campi di testo in formato libero, ad esempio un campo Name (Nome). Questo include quando si lavora con CloudWatch Logs o altri Servizi AWS tramite console, API, AWS CLI o SDK AWS. I dati inseriti nei tag o nei campi di testo in formato libero utilizzati per i nomi possono essere utilizzati per i log di fatturazione o di diagnostica. Quando fornisci un URL a un server esterno, ti suggeriamo vivamente di non includere informazioni sulle credenziali nell'URL per convalidare la tua richiesta al server.

Crittografia dei dati a riposo

CloudWatch Logs protegge i dati a riposo mediante crittografia Tutti i gruppi di log sono crittografati. Per impostazione predefinita, il servizio CloudWatch Logs gestisce le chiavi di crittografia lato server.

Se desideri gestire le chiavi usate per crittografare e decrittografare i log, utilizza le chiavi master cliente (CMK) da AWS Key Management Service. Per ulteriori informazioni, consulta Crittografa i dati di registro in CloudWatch Logs utilizzando AWS Key Management Service.

Crittografia dei dati in transito

CloudWatch Logs utilizza la crittografia end-to-end dei dati in transito. Il servizio CloudWatch Logs gestisce le chiavi di crittografia lato server.

Gestione delle identità e degli accessi per Amazon CloudWatch Logs

L'accesso ad Amazon CloudWatch Logs richiede credenziali che AWS possono essere utilizzate per autenticare le tue richieste. Tali credenziali devono disporre delle autorizzazioni per accedere alle AWS risorse, ad esempio per recuperare i dati di CloudWatch Logs relativi alle risorse cloud. Le seguenti sezioni forniscono dettagli su come utilizzare AWS Identity and Access Management (IAM) e CloudWatch Logs per proteggere le risorse controllando chi può accedervi:

- Autenticazione
- Controllo accessi

Autenticazione

Per fornire l'accesso, aggiungi autorizzazioni ai tuoi utenti, gruppi o ruoli:

- Utenti e gruppi in AWS IAM Identity Center:

  Crea un set di autorizzazioni. Segui le istruzioni riportate nella pagina Create a permission set (Creazione di un set di autorizzazioni) nella Guida per l'utente diAWS IAM Identity Center .

- Utenti gestiti in IAM tramite un provider di identità:

  Crea un ruolo per la federazione delle identità. Segui le istruzioni riportate nella pagina Creating a role for a third-party identity provider (federation) (Creazione di un ruolo per un provider di identità di terze parti [federazione]) nella Guida per l'utente di IAM.

- Utenti IAM:
  - Crea un ruolo che l'utente possa assumere. Per istruzioni, consulta la pagina Creating a role for an IAM user (Creazione di un ruolo per un utente IAM) nella Guida per l'utente di IAM.
  - (Non consigliato) Collega una policy direttamente a un utente o aggiungi un utente a un gruppo di utenti. Segui le istruzioni riportate nella pagina Aggiunta di autorizzazioni a un utente (console) nella Guida per l'utente di IAM.

Controllo accessi

Puoi disporre di credenziali valide per autenticare le tue richieste, ma a meno che tu non disponga delle autorizzazioni necessarie non puoi creare o accedere alle risorse di Logs. CloudWatch Ad esempio, è necessario disporre delle autorizzazioni per creare flussi di log, gruppi di log e così via.

Le sezioni seguenti descrivono come gestire le autorizzazioni per i registri. CloudWatch Consigliamo di leggere prima la panoramica.

- Panoramica della gestione delle autorizzazioni di accesso alle risorse Logs CloudWatch
- Utilizzo di politiche basate sull'identità (politiche IAM) per i registri CloudWatch
- CloudWatch Registra il riferimento alle autorizzazioni

Panoramica della gestione delle autorizzazioni di accesso alle risorse Logs CloudWatch

Per fornire l'accesso, aggiungi autorizzazioni ai tuoi utenti, gruppi o ruoli:

- Utenti e gruppi in: AWS IAM Identity Center

Crea un set di autorizzazioni. Segui le istruzioni riportate nella pagina Create a permission set (Creazione di un set di autorizzazioni) nella Guida per l'utente diAWS IAM Identity Center .

- Utenti gestiti in IAM tramite un provider di identità:

  Crea un ruolo per la federazione delle identità. Segui le istruzioni riportate nella pagina Creating a role for a third-party identity provider (federation) (Creazione di un ruolo per un provider di identità di terze parti [federazione]) nella Guida per l'utente di IAM.

- Utenti IAM:
  - Crea un ruolo che l'utente possa assumere. Per istruzioni, consulta la pagina Creating a role for an IAM user (Creazione di un ruolo per un utente IAM) nella Guida per l'utente di IAM.
  - (Non consigliato) Collega una policy direttamente a un utente o aggiungi un utente a un gruppo di utenti. Segui le istruzioni riportate nella pagina Aggiunta di autorizzazioni a un utente (console) nella Guida per l'utente di IAM.

Argomenti

CloudWatch Registra risorse e operazioni

In CloudWatch Logs le risorse principali sono i gruppi di log, i flussi di log e le destinazioni. CloudWatch Logs non supporta le risorse secondarie (altre risorse da utilizzare con la risorsa principale).

Alle risorse e alle risorse secondarie sono associati Amazon Resource Name (ARN) univoci, come illustrato nella tabella seguente.

Tipo di risorsa	Formato ARN
Gruppo di log	Si utilizzano entrambi i seguenti elementi. Il secondo, con la * fine, è ciò che viene restituito

Tipo di risorsa	Formato ARN
	dal comando `describe-log-groups` CLI e dall'DescribeLogGroupsAPI. arn:aws:logs:*region*:*account-id* :log-group:*log_group_name* arn:aws:logs:*region*:*account-id* :log-group:*log_group_name* :*
Flusso di log	*arn:aws:logs: region: account-id:log-group: log_group_name:log-stream: log-stream-name*
Destinazione	arn:aws:logs:*region*:*account-id* :destination:*destination_name*

Per ulteriori informazioni sugli ARN consulta ARN nella Guida per l'utente di IAM. Per informazioni sui CloudWatch Logs ARN, consulta Amazon Resource Names (ARNs) in. Riferimenti generali di Amazon Web Services Per un esempio di policy che copre i CloudWatch Logs, consulta. Utilizzo di politiche basate sull'identità (politiche IAM) per i registri CloudWatch

CloudWatch Logs fornisce una serie di operazioni per utilizzare le risorse CloudWatch Logs. Per un elenco di operazioni disponibili, consulta la sezione CloudWatch Registra il riferimento alle autorizzazioni.

Informazioni sulla proprietà delle risorse

L' AWS account possiede le risorse create nell'account, indipendentemente da chi ha creato le risorse. In particolare, il proprietario della risorsa è l' AWS account dell'entità principale (ovvero l'account root, un utente o un ruolo IAM) che autentica la richiesta di creazione delle risorse. Negli esempi seguenti viene illustrato il funzionamento:

- Se utilizzi le credenziali dell'account root del tuo AWS account per creare un gruppo di log, quest'ultimo è il AWS proprietario della CloudWatch risorsa Logs.
- Se crei un utente nel tuo AWS account e concedi le autorizzazioni per creare risorse CloudWatch Logs a quell'utente, l'utente può creare risorse Logs. CloudWatch Tuttavia, l' AWS account a cui appartiene l'utente è proprietario delle risorse Logs. CloudWatch

- Se crei un ruolo IAM nel tuo AWS account con le autorizzazioni per creare risorse CloudWatch Logs, chiunque possa assumere il ruolo può creare CloudWatch risorse Logs. Il tuo AWS account, a cui appartiene il ruolo, possiede le CloudWatch risorse Logs.

Gestione dell'accesso alle risorse

La policy delle autorizzazioni descrive chi ha accesso a cosa. Nella sezione seguente vengono descritte le opzioni disponibili per la creazione di policy relative alle autorizzazioni.

> **Note**
>
> Questa sezione illustra l'utilizzo di IAM nel contesto dei CloudWatch log. Non vengono fornite informazioni dettagliate sul servizio IAM. Per la documentazione di IAM completa, consulta la pagina Che cos'è IAM? nella Guida per l'utente di IAM. Per informazioni sulla sintassi delle policy IAM e le rispettive descrizioni, consultare Riferimento alle policy IAM di nella Guida per l'utente di IAM.

Le politiche associate a un'identità IAM sono denominate politiche basate sull'identità (politiche IAM) e le politiche allegate a una risorsa sono denominate politiche basate sulle risorse. CloudWatch Logs supporta politiche basate sull'identità e politiche basate sulle risorse per le destinazioni, utilizzate per abilitare sottoscrizioni tra account. Per ulteriori informazioni, consulta Condivisione di dati di log tra più account con le sottoscrizioni.

Argomenti

- Autorizzazioni del gruppo di log e Contributor Insights
- Policy basate su risorse

Autorizzazioni del gruppo di log e Contributor Insights

Contributor Insights è una funzionalità CloudWatch che consente di analizzare i dati dei gruppi di log e creare serie temporali che visualizzano i dati dei collaboratori. Puoi visualizzare i parametri relative ai primi N collaboratori, al numero totale di collaboratori univoci e al loro utilizzo. Per ulteriori informazioni, consulta Utilizzo di Contributor Insights per analizzare dati ad alta cardinalità.

Quando concedi a un utente le `cloudwatch:GetInsightRuleReport` autorizzazioni `cloudwatch:PutInsightRule` and, quell'utente può creare una regola che valuta qualsiasi gruppo di log in CloudWatch Logs e quindi visualizzare i risultati. I risultati possono contenere dati

dei collaboratori per tali gruppi di log. Assicurarsi di concedere queste autorizzazioni solo a utenti che devono essere in grado di visualizzare questi dati.

Policy basate su risorse

CloudWatch Logs supporta politiche basate sulle risorse per le destinazioni, che puoi utilizzare per abilitare gli abbonamenti tra più account. Per ulteriori informazioni, consulta Passaggio 1: creazione di una destinazione. Le destinazioni possono essere create utilizzando l'PutDestinationAPI ed è possibile aggiungere una politica delle risorse alla destinazione utilizzando l'API. PutDestination L'esempio seguente consente a un altro AWS account con l'ID account 111122223333 di iscrivere i propri gruppi di log alla destinazione. `arn:aws:logs:us-east-1:123456789012:destination:testDestination`

```
{
  "Version" : "2012-10-17",
  "Statement" : [
    {
      "Sid" : "",
      "Effect" : "Allow",
      "Principal" : {
        "AWS" : "111122223333"
      },
      "Action" : "logs:PutSubscriptionFilter",
      "Resource" : "arn:aws:logs:us-east-1:123456789012:destination:testDestination"
    }
  ]
}
```

Specifica degli elementi delle policy: operazioni, effetti e principali

Per ogni risorsa CloudWatch Logs, il servizio definisce un set di operazioni API. Per concedere le autorizzazioni per queste operazioni API, CloudWatch Logs definisce una serie di azioni che è possibile specificare in una politica. Alcune operazioni API possono richiedere le autorizzazioni per più di un'azione al fine di eseguire l'operazione API. Per ulteriori informazioni sulle risorse e sulle operazioni delle API, consulta CloudWatch Registra risorse e operazioni e CloudWatch Registra il riferimento alle autorizzazioni.

Di seguito sono elencati gli elementi di base di una policy:

- Risorsa - Usa un Amazon Resource Name (ARN) per identificare la risorsa a cui si applica la policy. Per ulteriori informazioni, consulta CloudWatch Registra risorse e operazioni.

- Operazione: utilizzi le parole chiave per identificare le operazioni sulla risorsa da permettere o rifiutare. Ad esempio, l'autorizzazione `logs.DescribeLogGroups` concede all'utente le autorizzazioni per eseguire l'operazione `DescribeLogGroups`.
- Effetto: specifica l'effetto, ovvero l'autorizzazione o il rifiuto, quando l'utente richiede l'operazione specifica. USe non concedi esplicitamente (consenti) l'accesso a una risorsa, l'accesso viene implicitamente rifiutato. Puoi anche rifiutare esplicitamente l'accesso a una risorsa per garantire che un utente non possa accedervi, anche se l'accesso viene concesso da un'altra policy.
- Principale: nelle policy basate su identità (policy IAM), l'utente a cui la policy è collegata è il principale implicito. Per le politiche basate sulle risorse, si specifica l'utente, l'account, il servizio o l'altra entità a cui si desidera ricevere le autorizzazioni (si applica solo alle politiche basate sulle risorse). CloudWatch Logs supporta politiche basate sulle risorse per le destinazioni.

Per ulteriori informazioni sulla sintassi e le descrizioni delle policy IAM, consulta AWS Riferimento alle policy IAM nella Guida per l'utente di IAM.

Per una tabella che mostra tutte le azioni dell'API CloudWatch Logs e le risorse a cui si applicano, consulta. CloudWatch Registra il riferimento alle autorizzazioni

Specifica delle condizioni in una policy

Quando concedi le autorizzazioni, puoi utilizzare la sintassi della policy di accesso per specificare le condizioni in base a cui la policy deve essere applicata. Ad esempio, potresti decidere che una policy venga applicata solo dopo una data specifica. Per ulteriori informazioni su come specificare le condizioni in un linguaggio di policy, consulta la sezione Condizione nella Guida per l'utente di IAM.

Per esprimere le condizioni è necessario utilizzare chiavi di condizione predefinite. Per un elenco delle chiavi di contesto supportate da ogni AWS servizio e un elenco di chiavi di policy a AWSlivello globale, consulta Azioni, risorse e chiavi di condizione per AWS i servizi e le chiavi di contesto delle condizioniAWS globali.

> **Note**
>
> È possibile utilizzare i tag per controllare l'accesso alle risorse di CloudWatch Logs, inclusi i gruppi di log e le destinazioni. L'accesso ai flussi di log è controllato a livello di gruppo di log per via della relazione gerarchica tra i gruppi di log e i flussi di log. Per ulteriori informazioni sull'utilizzo dei tag per controllare l'accesso alle risorse, consulta Controllo dell'accesso alle risorse di Amazon Web Services utilizzando i tag.

Utilizzo di politiche basate sull'identità (politiche IAM) per i registri CloudWatch

In questo argomento vengono forniti esempi di policy basate su identità in cui un amministratore account può collegare policy di autorizzazione a identità IAM, ovvero utenti, gruppi e ruoli.

> **Important**
>
> Ti consigliamo di consultare innanzitutto gli argomenti introduttivi che spiegano i concetti e le opzioni di base disponibili per gestire l'accesso alle tue risorse Logs. CloudWatch Per ulteriori informazioni, consulta Panoramica della gestione delle autorizzazioni di accesso alle risorse Logs CloudWatch .

Questo argomento comprende quanto segue:

- Autorizzazioni necessarie per utilizzare la console CloudWatch
- AWS politiche gestite (predefinite) per i registri CloudWatch
- Esempi di policy gestite dal cliente

Di seguito è riportato un esempio di policy delle autorizzazioni:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents",
        "logs:DescribeLogStreams"
    ],
      "Resource": [
        "arn:aws:logs:*:*:*"
    ]
  }
 ]
}
```

Questa policy dispone di una dichiarazione che concede autorizzazioni per creare gruppi e flussi di log, caricare eventi di log ai flussi di log ed elencare dettagli relativi ai flussi di log.

Il carattere jolly (*) alla fine del valore `Resource` indica che la dichiarazione concede l'autorizzazione per le operazioni `logs:CreateLogGroup`, `logs:CreateLogStream`, `logs:PutLogEvents` e `logs:DescribeLogStreams` su qualsiasi gruppo di log. Per limitare questa autorizzazione a uno specifico gruppo di log, sostituisci il carattere jolly (*) nell'ARN della risorsa con l'ARN del gruppo di log specifico. Per ulteriori informazioni sulle sezioni all'interno della dichiarazione di policy IAM, consulta Riferimento agli elementi di policy IAM nella Guida per l'utente di IAM. Per un elenco che mostra tutte le azioni di CloudWatch Logs, consulta. CloudWatch Registra il riferimento alle autorizzazioni

Autorizzazioni necessarie per utilizzare la console CloudWatch

Affinché un utente possa utilizzare CloudWatch Logs nella CloudWatch console, deve disporre di un set minimo di autorizzazioni che gli consentano di descrivere AWS le altre risorse del proprio account. AWS Per utilizzare CloudWatch Logs nella CloudWatch console, è necessario disporre delle autorizzazioni dei seguenti servizi:

- CloudWatch
- CloudWatch Registri
- OpenSearch Servizio
- IAM
- Kinesis
- Lambda
- Amazon S3

Se decidi di creare una policy IAM più restrittiva delle autorizzazioni minime richieste, la console non funzionerà come previsto per gli utenti con tale policy IAM. Per garantire che tali utenti possano continuare a utilizzare la CloudWatch console, allega anche la policy `CloudWatchReadOnlyAccess` gestita all'utente, come descritto inAWS politiche gestite (predefinite) per i registri CloudWatch .

Non è necessario consentire autorizzazioni minime per la console per gli utenti che effettuano chiamate solo verso l'API AWS CLI o verso l'API CloudWatch Logs.

Il set completo di autorizzazioni necessarie per lavorare con la CloudWatch console per un utente che non utilizza la console per gestire gli abbonamenti ai registri è:

- cloudwatch: GetMetricData
- orologio nuvoloso: ListMetrics
- registri: CancelExportTask
- registri: CreateExportTask
- registri: CreateLogGroup
- registri: CreateLogStream
- registri: DeleteLogGroup
- registri: DeleteLogStream
- registri: DeleteMetricFilter
- registri: DeleteQueryDefinition
- registri: DeleteRetentionPolicy
- registri: DeleteSubscriptionFilter
- registri: DescribeExportTasks
- registri: DescribeLogGroups
- registri: DescribeLogStreams
- registri: DescribeMetricFilters
- registri: DescribeQueryDefinitions
- registri: DescribeQueries
- registri: DescribeSubscriptionFilters
- registri: FilterLogEvents
- registri: GetLogEvents
- registri: GetLogGroupFields
- registri: GetLogRecord
- registri: GetQueryResults
- registri: PutMetricFilter
- registri: PutQueryDefinition
- registri: PutRetentionPolicy

- registri: StartQuery
- registri: StopQuery
- registri: PutSubscriptionFilter
- registri: TestMetricFilter

Per un utente che intende utilizzare anche la console per gestire le sottoscrizioni ai log, sono necessarie anche le seguenti autorizzazioni:

- Sì: DescribeElasticsearchDomain
- Sì: ListDomainNames
- sono: AttachRolePolicy
- sono: CreateRole
- sono: GetPolicy
- sono: GetPolicyVersion
- sono: GetRole
- sono: ListAttachedRolePolicies
- sono: ListRoles
- cinesi: DescribeStreams
- cinesi: ListStreams
- lambda: AddPermission
- lambda: CreateFunction
- lambda: GetFunctionConfiguration
- lambda: ListAliases
- lambda: ListFunctions
- lambda: ListVersionsByFunction
- lambda: RemovePermission
- s3: ListBuckets

AWS politiche gestite (predefinite) per i registri CloudWatch

AWS affronta molti casi d'uso comuni fornendo politiche IAM autonome create e amministrate da. AWSLe policy gestite concedono le autorizzazioni necessarie per i casi di utilizzo comune in modo

da non dover cercare quali sono le autorizzazioni richieste. Per ulteriori informazioni, consulta Policy gestite daAWS nella Guida per l'utente di IAM.

Le seguenti politiche AWS gestite, che puoi allegare agli utenti e ai ruoli del tuo account, sono specifiche dei CloudWatch log:

- CloudWatchLogsFullAccess— Garantisce l'accesso completo ai registri. CloudWatch
- CloudWatchLogsReadOnlyAccess— Garantisce l'accesso in sola lettura ai registri. CloudWatch

CloudWatchLogsFullAccess

La CloudWatchLogsFullAccesspolitica garantisce l'accesso completo ai registri. CloudWatch I contenuti sono come indicato di seguito:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "logs:*"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

CloudWatchLogsReadOnlyAccess

La CloudWatchLogsReadOnlyAccesspolitica garantisce l'accesso in sola lettura ai registri. CloudWatch I contenuti sono come indicato di seguito:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "logs:Describe*",
                "logs:Get*",
                "logs:List*",
```

```
                "logs:StartQuery",
                "logs:StopQuery",
                "logs:TestMetricFilter",
                "logs:FilterLogEvents",
                "logs:StartLiveTail",
                "logs:StopLiveTail"
            ],
            "Resource": "*"
        }
    ]
}
```

CloudWatchLogsCrossAccountSharingConfiguration

La CloudWatchLogsCrossAccountSharingConfigurationpolitica consente l'accesso alla creazione, alla gestione e alla visualizzazione dei collegamenti di Observability Access Manager per la condivisione delle risorse di Logs tra account. CloudWatch Per ulteriori informazioni, consulta CloudWatch osservabilità tra account.

I contenuti sono come indicato di seguito:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "logs:Link",
                "oam:ListLinks"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "oam:DeleteLink",
                "oam:GetLink",
                "oam:TagResource"
            ],
            "Resource": "arn:aws:oam:*:*:link/*"
        },
        {
            "Effect": "Allow",
```

```
            "Action": [
                "oam:CreateLink",
                "oam:UpdateLink"
            ],
            "Resource": [
                "arn:aws:oam:*:*:link/*",
                "arn:aws:oam:*:*:sink/*"
            ]
        }
    ]
}
```

CloudWatch Registra gli aggiornamenti delle politiche gestite AWS

Visualizza i dettagli sugli aggiornamenti alle politiche AWS gestite per CloudWatch Logs da quando questo servizio ha iniziato a tenere traccia di queste modifiche. Per ricevere avvisi automatici sulle modifiche a questa pagina, iscriviti al feed RSS nella pagina della cronologia dei documenti di CloudWatch Logs.

Modifica	Descrizione	Data
CloudWatchLogsRead OnlyAccess: aggiornamento a una policy esistente	CloudWatch Registra le autorizzazioni aggiunte a. CloudWatchLogsRead OnlyAccess Le `logs:StopLiveTail` autorizzazioni `logs:StartLiveTail` e sono state aggiunte in modo che gli utenti con questo criterio possano utilizzare la console per avviare e interrompere le sessioni live tail di CloudWatch Logs. Per ulteriori informazioni, consulta Use live tail to view logs in near real time.	6 giugno 2023

Modifica	Descrizione	Data
CloudWatchLogsCrossAccountSharingConfiguration: nuova policy	CloudWatch Logs ha aggiunto una nuova politica che consente di gestire i link di osservabilità CloudWatch tra account che condividono i gruppi di log di Logs. CloudWatch Per ulteriori informazioni, consulta osservabilità tra account CloudWatch	27 novembre 2022
CloudWatchLogsFullAccess: aggiornamento a una policy esistente	CloudWatch Registra le autorizzazioni aggiunte a. CloudWatchLogsFullAccess Le `oam:ListAttachedLinks` autorizzazioni `oam:ListSinks` e sono state aggiunte in modo che gli utenti con questo criterio possano utilizzare la console per visualizzare i dati condivisi dagli account di origine in CloudWatch modo osservabile tra più account.	27 novembre 2022

Modifica	Descrizione	Data
CloudWatchLogsReadOnlyAccess: aggiornamento a una policy esistente	CloudWatch Registra le autorizzazioni aggiunte a. CloudWatchLogsReadOnlyAccess Le `oam:ListAttachedLinks` autorizzazioni `oam:ListSinks` e sono state aggiunte in modo che gli utenti con questo criterio possano utilizzare la console per visualizzare i dati condivisi dagli account di origine in CloudWatch modo osservabile tra più account.	27 novembre 2022

Esempi di policy gestite dal cliente

Puoi creare politiche IAM personalizzate per consentire le autorizzazioni per le azioni e le risorse di CloudWatch Logs. Puoi collegare queste policy personalizzate agli utenti o ai gruppi che richiedono le autorizzazioni.

In questa sezione, puoi trovare esempi di politiche utente che concedono autorizzazioni per varie CloudWatch azioni di Logs. Queste politiche funzionano quando utilizzi l'API CloudWatch Logs, gli AWS SDK o il. AWS CLI

Esempi

- Esempio 1: consenti l'accesso completo ai log CloudWatch
- Esempio 2: consentire l'accesso in sola lettura ai registri CloudWatch
- Esempio 3: consentire l'accesso a un gruppo di log

Esempio 1: consenti l'accesso completo ai log CloudWatch

La seguente politica consente a un utente di accedere a tutte le azioni di CloudWatch Logs.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "logs:*"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

Esempio 2: consentire l'accesso in sola lettura ai registri CloudWatch

AWS fornisce una CloudWatchLogsReadOnlyAccesspolitica che consente l'accesso in sola lettura ai dati dei registri. CloudWatch Questa policy include le seguenti autorizzazioni:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "logs:Describe*",
                "logs:Get*",
                "logs:List*",
                "logs:StartQuery",
                "logs:StopQuery",
                "logs:TestMetricFilter",
                "logs:FilterLogEvents",
                "logs:StartLiveTail",
                "logs:StopLiveTail"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

Esempio 3: consentire l'accesso a un gruppo di log

La policy seguente consente a un utente di leggere e scrivere eventi di log in un gruppo di log specificato.

> **Important**
>
> I caratteri `:*` alla fine del nome del gruppo di log sulla riga `Resource` sono necessari per indicare che la policy si applica a tutti i flussi di log in questo gruppo di log. Se ometti `:*`, la policy non verrà applicata.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
      "Action": [
        "logs:CreateLogStream",
        "logs:DescribeLogStreams",
        "logs:PutLogEvents",
        "logs:GetLogEvents"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:logs:us-west-2:123456789012:log-group:SampleLogGroupName:*"
      }
   ]
}
```

Utilizzo del tagging e delle policy IAM per il controllo a livello di gruppo di log

È possibile concedere agli utenti l'accesso a determinati gruppi di log evitando che accedano ad altri gruppi di log. Per farlo, aggiungere un tag ai gruppi di log e usare policy IAM che fanno riferimento a tali tag. Per applicare i tag a un gruppo di log, è necessario disporre dell'autorizzazione `logs:TagResource` o `logs:TagLogGroup`. Ciò vale sia se si assegnano tag al gruppo di log al momento della creazione, sia se li si assegna successivamente.

Per ulteriori informazioni sull'applicazione di tag ai gruppi di log, consulta Contrassegna i gruppi di log in Amazon CloudWatch Logs.

Quando si aggiunge un tag ai gruppi di log, è possibile concedere una policy IAM a un utente per consentire l'accesso solo ai gruppi di log con un determinato tag. Ad esempio, la seguente istruzione di policy garantisce l'accesso solo ai gruppi di log con il valore `Green` per la chiave di tag `Team`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "logs:*"
            ],
            "Effect": "Allow",
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "aws:ResourceTag/Team": "Green"
                }
            }
        }
    ]
}
```

Le operazioni StopQuerye le StopLiveTailAPI non interagiscono con AWS le risorse nel senso tradizionale. Non restituiscono né inseriscono alcun dato, né modificano una risorsa in alcun modo. Funzionano invece solo su una determinata sessione live tail o su una determinata query di CloudWatch Logs Insights, che non sono classificate come risorse. Di conseguenza, quando per queste operazioni si specifica il campo `Resource` nelle policy IAM, è necessario impostare il valore del campo `Resource` come *, analogamente all'esempio di seguito.

```
{
    "Version": "2012-10-17",
    "Statement":
        [ {
            "Effect": "Allow",
            "Action": [
                "logs:StopQuery",
                "logs:StopLiveTail"
            ],
            "Resource": "*"
            }
        ]
```

```
}
```

Per ulteriori informazioni sull'utilizzo di istruzioni di policy IAM, consulta Controllo dell'accesso tramite le policy nella Guida per l'utente di IAM.

CloudWatch Registra il riferimento alle autorizzazioni

Quando configuri Controllo accessi e scrivi policy di autorizzazioni che puoi collegare a un'identità IAM (policy basate su identità), puoi usare la tabella seguente come riferimento. La tabella elenca ogni operazione dell'API CloudWatch Logs e le azioni corrispondenti per le quali è possibile concedere le autorizzazioni per eseguire l'azione. Puoi specificare le operazioni nel campo `Action` della policy. Per il `Resource` campo, è possibile specificare l'ARN di un gruppo di log o di un flusso di log oppure specificare di `*` rappresentare tutte le risorse CloudWatch Logs.

È possibile utilizzare le chiavi di condizione AWS-wide nelle politiche di CloudWatch Logs per esprimere condizioni. Per un elenco completo delle chiavi AWS-wide, consulta AWS Global and IAM Condition Context Keys nella IAM User Guide.

> **Note**
>
> Per specificare un'operazione, utilizza il prefisso `logs:` seguito dal nome dell'operazione API. Ad esempio: `logs:CreateLogGrouplogs:CreateLogStream`, o `logs:*` (per tutte le azioni di CloudWatch Logs).

CloudWatch Registra le operazioni API e le autorizzazioni richieste per le azioni

CloudWatch Registra le operazioni API	Autorizzazioni necessarie (operazioni API)
CancelExportTask	`logs:CancelExportTask` Necessaria per eliminare un'attività di esportazione in esecuzione o pendente.
CreateExportTask	`logs:CreateExportTask` Necessaria per esportare dati da un gruppo di log a un bucket Amazon S3.
CreateLogGroup	`logs:CreateLogGroup`

CloudWatch Registra le operazioni API	Autorizzazioni necessarie (operazioni API)
	Necessaria per creare un nuovo gruppo di log.
CreateLogStream	`logs:CreateLogStream` Necessaria per creare un nuovo flusso di log in un gruppo di log.
DeleteDestination	`logs:DeleteDestination` Necessaria per eliminare una destinazione di log e disabilita tutti i filtri di sottoscrizione.
DeleteLogGroup	`logs:DeleteLogGroup` Necessario per eliminare un gruppo di log e tutti i log eventi archiviati associati.
DeleteLogStream	`logs:DeleteLogStream` Necessaria per eliminare un flusso di log e tutti gli eventi di log archiviati associati.
DeleteMetricFilter	`logs:DeleteMetricFilter` Necessaria per eliminare un filtro parametri associato a un gruppo di log.
DeleteQueryDefinition	`logs:DeleteQueryDefinition` Necessario per eliminare una definizione di query salvata in CloudWatch Logs Insights.
DeleteResourcePolicy	`logs:DeleteResourcePolicy` Necessario per eliminare una politica delle risorse CloudWatch di Logs.

CloudWatch Registra le operazioni API	Autorizzazioni necessarie (operazioni API)
DeleteRetentionPolicy	`logs:DeleteRetentionPolicy` Necessaria per eliminare una policy di retention di un gruppo di log.
DeleteSubscriptionFilter	`logs:DeleteSubscriptionFilter` Necessaria per eliminare un filtro sottoscrizioni associato a un gruppo di log.
DescribeDestinations	`logs:DescribeDestinations` Necessaria per visualizzare tutte le destinazioni associate all'account.
DescribeExportTasks	`logs:DescribeExportTasks` Necessaria per visualizzare tutte le attività di esportazione associate all'account.
DescribeLogGroups	`logs:DescribeLogGroups` Necessaria per visualizzare tutti i gruppi di log associati all'account.
DescribeLogStreams	`logs:DescribeLogStreams` Necessaria per visualizzare tutti i flussi di log associati a un gruppo di log.
DescribeMetricFilters	`logs:DescribeMetricFilters` Necessario per visualizzare tutti i parametri associati a un gruppo di log.

CloudWatch Registra le operazioni API	Autorizzazioni necessarie (operazioni API)
DescribeQueryDefinitions	`logs:DescribeQueryDefinitions` Necessario per visualizzare l'elenco delle definizioni delle query salvate in CloudWatch Logs Insights.
DescribeQueries	`logs:DescribeQueries` Necessario per visualizzare l'elenco delle query di CloudWatch Logs Insights pianificate, in esecuzione o eseguite di recente.
DescribeResourcePolicies	`logs:DescribeResourcePolicies` Necessario per visualizzare un elenco delle politiche relative alle risorse di Logs. CloudWatch
DescribeSubscriptionFilters	`logs:DescribeSubscriptionFilters` Necessaria per visualizzare tutti i filtri di sottoscrizione associati a un gruppo di log.
FilterLogEvents	`logs:FilterLogEvents` Necessaria per ordinare log eventi in base a un modello filtro di gruppo di log.
GetLogEvents	`logs:GetLogEvents` Necessaria per recuperare eventi di log da un flusso di log.
GetLogGroupFields	`logs:GetLogGroupFields` Necessaria per recuperare l'elenco dei campi inclusi negli eventi di log in un gruppo di log.

CloudWatch Registra le operazioni API	Autorizzazioni necessarie (operazioni API)
GetLogRecord	`logs:GetLogRecord` Necessaria per recuperare i dettagli da un singolo log eventi.
GetQueryResults	`logs:GetQueryResults` Necessario per recuperare i risultati delle interrogazioni di CloudWatch Logs Insights.
ListTagsLogGroup	`logs:ListTagsLogGroup` Necessaria per elencare i tag associati a un gruppo di log.
PutDestination	`logs:PutDestination` Necessaria per creare o aggiornare un flusso di log di destinazione (ad esempio, un flusso Kinesis).
PutDestinationPolicy	`logs:PutDestinationPolicy` Necessaria per creare o aggiornare una policy di accesso predefinita associata a una destinazi one di log esistente.
PutLogEvents	`logs:PutLogEvents` Necessaria per caricare un batch di eventi di log in un flusso di eventi.
PutMetricFilter	`logs:PutMetricFilter` Necessaria per creare o aggiornare un filtro di parametri e associarlo a un gruppo di log.

CloudWatch Registra le operazioni API	Autorizzazioni necessarie (operazioni API)
PutQueryDefinition	`logs:PutQueryDefinition` Necessario per salvare una query in CloudWatch Logs Insights.
PutResourcePolicy	`logs:PutResourcePolicy` Necessario per creare una politica delle risorse CloudWatch di Logs.
PutRetentionPolicy	`logs:PutRetentionPolicy` Necessaria per impostare il numero di giorni per conservare log eventi (retention) in un gruppo di log.
PutSubscriptionFilter	`logs:PutSubscriptionFilter` Necessaria per creare o aggiornare un filtro sottoscrizioni e associarlo a un gruppo di log.
StartQuery	`logs:StartQuery` Necessario per avviare le query di CloudWatch Logs Insights.
StopQuery	`logs:StopQuery` Necessario per interrompere una query di CloudWatch Logs Insights in corso.
TagLogGroup	`logs:TagLogGroup` Necessaria per aggiungere o aggiornare i tag dei gruppi di log.

CloudWatch Registra le operazioni API	Autorizzazioni necessarie (operazioni API)
TestMetricFilter	`logs:TestMetricFilter` Necessaria per verificare un modello di filtro su un campionamento di messaggi di log eventi.

Utilizzo di ruoli collegati ai servizi per i registri CloudWatch

Amazon CloudWatch Logs utilizza ruoli collegati ai servizi AWS Identity and Access Management (IAM). Un ruolo collegato ai servizi è un tipo unico di ruolo IAM collegato direttamente ai log. CloudWatch I ruoli collegati ai servizi sono predefiniti da CloudWatch Logs e includono tutte le autorizzazioni richieste dal servizio per chiamare altri servizi per tuo conto. AWS

Un ruolo collegato al servizio rende la configurazione di CloudWatch Logs più efficiente perché non è necessario aggiungere manualmente le autorizzazioni necessarie. CloudWatch Logs definisce le autorizzazioni dei ruoli collegati ai servizi e, se non diversamente definito, solo Logs può assumere tali ruoli. CloudWatch Le autorizzazioni definite includono policy di attendibilità e di autorizzazioni. Queste ultime non possono essere collegate a nessun'altra entità IAM.

Per informazioni sugli altri servizi che supportano i ruoli collegati ai servizi consulta ServiziAWS che funzionano con IAM. Cerca i servizi che hanno Sì nella colonna Ruolo collegato ai servizi. Scegli Sì in corrispondenza di un link per visualizzare la documentazione relativa al ruolo collegato ai servizi per tale servizio.

Autorizzazioni relative ai ruoli collegati al servizio per Logs CloudWatch

CloudWatch Logs utilizza il ruolo collegato al servizio denominato. AWSServiceRoleForLogDelivery CloudWatch Logs utilizza questo ruolo collegato al servizio per scrivere i log direttamente su Kinesis Data Firehose. Per ulteriori informazioni, consulta Abilitazione della registrazione dai servizi AWS.

Ai fini dell'assunzione del ruolo, il ruolo collegato ai servizi AWSServiceRoleForLogDelivery considera attendibili i seguenti servizi:

- `logs.amazonaws.com`

La politica di autorizzazione dei ruoli consente a CloudWatch Logs di completare le seguenti azioni sulle risorse specificate:

- Operazione: `firehose:PutRecord` e `firehose:PutRecordBatch` su tutti i flussi Kinesis Data Firehose che hanno un tag con una chiave `LogDeliveryEnabled` con un valore di `True`. Questo tag viene automaticamente allegato a un flusso Kinesis Data Firehose quando si crea una sottoscrizione per consegnare i log a Kinesis Data Firehose.

Per consentire a un'entità IAM, di creare, modificare o eliminare un ruolo collegato ai servizi, devi configurare le autorizzazioni. Questa entità può essere un utente, un gruppo o un ruolo. Per ulteriori informazioni, consulta Autorizzazioni del ruolo collegato ai servizi nella Guida per l'utente di IAM.

Creazione di un ruolo collegato al servizio per Logs CloudWatch

Non è necessario creare manualmente un ruolo collegato ai servizi. Quando configuri i log da inviare direttamente a uno stream Kinesis Data Firehose nell'API, nella o AWS Management Consolenell'API CloudWatch , AWS CLILogs crea AWS automaticamente il ruolo collegato al servizio.

Se elimini questo ruolo collegato ai servizi, puoi ricrearlo seguendo lo stesso processo utilizzato per ricreare il ruolo nell'account. Quando configuri nuovamente i log per essere inviati direttamente a uno stream Kinesis Data Firehose CloudWatch , Logs crea nuovamente il ruolo collegato al servizio per te.

Modifica di un ruolo collegato al servizio per Logs CloudWatch

CloudWatch I registri non consentono di modificare AWSServiceRoleForLogDelivery, o qualsiasi altro ruolo collegato al servizio, dopo averlo creato. Non è possibile modificare il nome del ruolo poiché varie entità possono farvi riferimento. Tuttavia, utilizzando IAM è possibile modificarne la descrizione. Per ulteriori informazioni, consulta la sezione Modifica di un ruolo collegato ai servizi nella Guida per l'utente di IAM.

Eliminazione di un ruolo collegato al servizio per Logs CloudWatch

Se non è più necessario utilizzare una funzionalità o un servizio che richiede un ruolo collegato al servizio, ti consigliamo di eliminare il ruolo. In questo modo non sarà più presente un'entità non utilizzata che non viene monitorata e gestita attivamente. Tuttavia, è necessario effettuare la pulizia delle risorse associate al ruolo collegato al servizio prima di poterlo eliminare manualmente.

> **Note**
>
> Se il servizio CloudWatch Logs utilizza il ruolo quando si tenta di eliminare le risorse, l'eliminazione potrebbe non riuscire. In questo caso, attendi alcuni minuti e quindi ripeti l'operazione.

Per eliminare le risorse CloudWatch Logs utilizzate dal ruolo collegato al servizio AWSServiceRoleForLogDelivery

- Interrompere l'invio dei log direttamente ai flussi Kinesis Data Firehose.

Per eliminare manualmente il ruolo collegato ai servizi mediante IAM

Utilizza la console IAM AWS CLI, o l' AWS API per eliminare il ruolo collegato al AWSServiceRoleForLogDeliveryservizio. Per ulteriori informazioni, consulta Eliminazione del ruolo collegato ai servizi.

Regioni supportate per i ruoli collegati al servizio CloudWatch Logs

CloudWatch Logs supporta l'utilizzo di ruoli collegati al servizio in tutte le AWS regioni in cui il servizio è disponibile. Per ulteriori informazioni, consulta CloudWatch Registra regioni ed endpoint.

Convalida della conformità per Amazon Logs CloudWatch

I revisori di terze parti valutano la sicurezza e la conformità di Amazon CloudWatch Logs nell'ambito di diversi programmi di AWS conformità. Questi includono SOC, PCI, FedRAMP, HIPAA e altri.

Per un elenco dei AWS servizi che rientrano nell'ambito di specifici programmi di conformità, consulta AWS Services in Scope by Compliance ProgramAWS Program. Per informazioni generali, consulta Programmi di conformitàAWS.

È possibile scaricare report di audit di terze parti utilizzando AWS Artifact. Per ulteriori informazioni, consulta Scaricamento dei report in AWS Artifact .

La tua responsabilità di conformità quando utilizzi Amazon CloudWatch Logs è determinata dalla sensibilità dei tuoi dati, dagli obiettivi di conformità della tua azienda e dalle leggi e dai regolamenti applicabili. AWS fornisce le seguenti risorse per contribuire alla conformità:

- Security and Compliance Quick Start Guides (Guide Quick Start Sicurezza e compliance): queste guide alla distribuzione illustrano considerazioni relative all'architettura e forniscono procedure per la distribuzione di ambienti di base incentrati sulla sicurezza e sulla conformità su AWS.
- Progettazione per la sicurezza e la conformità HIPAA su Amazon Web Services: questo white paper descrive come le aziende possono utilizzare AWS per creare applicazioni conformi allo standard HIPAA.
- AWS Risorse per la conformità Risorse perAWS : questa raccolta di cartelle di lavoro e guide potrebbe riguardare il tuo settore e la tua area geografica.
- Evaluating Resources with Rules nella AWS Config Developer Guide: AWS Configvaluta la conformità delle configurazioni delle risorse alle pratiche interne, alle linee guida del settore e alle normative.
- AWS Security Hub— Questo AWS servizio offre una visione completa dello stato di sicurezza dell'utente e consente di verificare la conformità agli standard e alle best practice del settore della sicurezza. AWS

Resilienza in Amazon CloudWatch Logs

L'infrastruttura globale di AWS è basata su regioni e zone di disponibilità AWS. Le regioni forniscono più zone di disponibilità fisicamente separate e isolate, connesse tramite reti altamente ridondanti, a bassa latenza e throughput elevato. Con le zone di disponibilità, è possibile progettare e gestire applicazioni e database che eseguono il failover automatico tra zone di disponibilità senza interruzioni. Le zone di disponibilità sono più disponibili, tolleranti ai guasti e scalabili rispetto alle infrastrutture a data center singolo o multiplo.

Per ulteriori informazioni sulle regioni e le zone di disponibilità AWS, consulta Infrastruttura globale di AWS.

Sicurezza dell'infrastruttura in Amazon CloudWatch Logs

In quanto servizio gestito, Amazon CloudWatch Logs è protetto dalla sicurezza di rete AWS globale. Per informazioni sui servizi di AWS sicurezza e su come AWS protegge l'infrastruttura, consulta AWS Cloud Security. Per progettare il tuo AWS ambiente utilizzando le migliori pratiche per la sicurezza dell'infrastruttura, vedi Infrastructure Protection in Security Pillar AWS Well-Architected Framework.

Si utilizzano chiamate API AWS pubblicate per accedere ai CloudWatch log attraverso la rete. I client devono supportare quanto segue:

- Transport Layer Security (TLS). È richiesto TLS 1.2 ed è consigliato TLS 1.3.
- Suite di cifratura con Perfect Forward Secrecy (PFS), ad esempio Ephemeral Diffie-Hellman (DHE) o Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). La maggior parte dei sistemi moderni, come Java 7 e versioni successive, supporta tali modalità.

Inoltre, le richieste devono essere firmate utilizzando un ID chiave di accesso e una chiave di accesso segreta associata a un principale IAM. In alternativa, è possibile utilizzare AWS Security Token Service (AWS STS) per generare le credenziali di sicurezza temporanee per sottoscrivere le richieste.

Utilizzo dei CloudWatch log con endpoint VPC di interfaccia

Se utilizzi Amazon Virtual Private Cloud (Amazon VPC) per ospitare AWS le tue risorse, puoi stabilire una connessione privata tra il tuo VPC e Logs. CloudWatch Puoi utilizzare questa connessione per inviare log a CloudWatch Logs senza inviarli tramite Internet.

Amazon VPC è un AWS servizio che puoi utilizzare per avviare AWS risorse in una rete virtuale definita dall'utente. Con un VPC;, detieni il controllo delle impostazioni della rete, come l'intervallo di indirizzi IP, le sottoreti, le tabelle di routing e i gateway di rete. Per connettere il tuo VPC ai CloudWatch log, definisci un endpoint VPC di interfaccia per Logs. CloudWatch Questo tipo di endpoint consente di collegare il VPC ai servizi AWS . L'endpoint fornisce una connettività affidabile e scalabile ai CloudWatch log senza richiedere un gateway Internet, un'istanza NAT (Network Address Translation) o una connessione VPN. Per ulteriori informazioni, consulta Che cos'è Amazon VPC? nella Guida per l'utente di Amazon VPC.

Gli endpoint VPC di interfaccia sono alimentati da AWS PrivateLink, una AWS tecnologia che consente la comunicazione privata tra AWS i servizi utilizzando un'interfaccia di rete elastica con indirizzi IP privati. Per ulteriori informazioni, vedere New — AWS PrivateLink for AWS Services.

Le fasi seguenti sono per gli utenti Amazon VPC. Per ulteriori informazioni, consulta l'argomento relativo alle nozioni di base nella Guida per l'utente di Amazon VPC.

Disponibilità

CloudWatch Logs attualmente supporta gli endpoint VPC in AWS tutte le regioni, incluse le regioni. AWS GovCloud (US)

Creazione di un endpoint VPC per i log CloudWatch

Per iniziare a utilizzare CloudWatch Logs con il tuo VPC, crea un endpoint VPC di interfaccia per Logs. CloudWatch Il servizio da scegliere è com.amazonaws.***Region***.logs. Non è necessario modificare alcuna impostazione per Logs. CloudWatch Per ulteriori informazioni, consulta Creazione di un endpoint di interfaccia nella Guida per l'utente di Amazon VPC.

Verifica della connessione tra il tuo VPC e Logs CloudWatch

Dopo aver creato l'endpoint, è possibile testare la connessione.

Per testare la connessione tra il tuo VPC e l'endpoint Logs CloudWatch

1. Connettiti a un'istanza Amazon EC2 che risiede nel tuo VPC. Per informazioni sulla connessione, consulta Connessione all'istanza Linux o Connessione all'istanza Windows nella documentazione Amazon EC2.
2. Dall'istanza, usa AWS CLI per creare una voce di registro in uno dei gruppi di log esistenti.

   Prima di tutto, creare un file JSON con un evento di log. Il timestamp deve essere specificato come numero di millisecondi dopo il 1° gennaio 1970 alle 00:00:00 UTC.

   ```
   [
     {
       "timestamp": 1533854071310,
       "message": "VPC Connection Test"
     }
   ]
   ```

   Quindi, utilizzare il comando `put-log-events` per creare la voce di log:

   ```
   aws logs put-log-events --log-group-name LogGroupName --log-stream-
   name LogStreamName --log-events file://JSONFileName
   ```

   Se la risposta al comando include `nextSequenceToken`, il comando è riuscito e l'endpoint VPC sta funzionando.

Controllo dell'accesso all'endpoint VPC CloudWatch Logs

Una policy endpoint VPC è una policy della risorsa IAM che viene collegata a un endpoint durante la creazione o la modifica dell'endpoint. Se non colleghi una policy durante la creazione di un endpoint, viene collegata una policy predefinita che consente l'accesso completo al servizio. Una policy endpoint non esclude né sostituisce policy IAM o policy specifiche del servizio. Si tratta di una policy separata per controllare l'accesso dall'endpoint al servizio specificato.

Le policy endpoint devono essere scritte in formato JSON.

Per ulteriori informazioni, consultare Controllo degli accessi ai servizi con endpoint VPC nella Guida per l'utente di Amazon VPC.

Di seguito è riportato un esempio di policy sugli endpoint per Logs. CloudWatch Questa policy consente agli utenti che si connettono ai CloudWatch log tramite il VPC di creare flussi di log e inviare log CloudWatch ai log, e impedisce loro di eseguire altre azioni di log. CloudWatch

```
{
  "Statement": [
    {
      "Sid": "PutOnly",
      "Principal": "*",
      "Action": [
        "logs:CreateLogStream",
        "logs:PutLogEvents"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

Per modificare la policy degli endpoint VPC per Logs CloudWatch

1. Accedere alla console Amazon VPC all'indirizzo https://console.aws.amazon.com/vpc/.
2. Nel pannello di navigazione, seleziona Endpoints (Endpoint).
3. Se non hai ancora creato l'endpoint for CloudWatch Logs, scegli Crea endpoint. Quindi seleziona com.amazonaws.***Region***.logs e scegli Create endpoint (Crea endpoint).
4. Seleziona l'endpoint com.amazonaws.***Region***.logs e scegli la scheda Policy nella parte inferiore dello schermo.

5. Scegli Edit Policy (Modifica policy) e apporta le modifiche alla policy.

Supporto delle chiavi di contesto VPC

CloudWatch Logs supporta `aws:SourceVpc` le chiavi di `aws:SourceVpce` contesto che possono limitare l'accesso a VPC o endpoint VPC specifici. Queste chiavi funzionano solo se l'utente utilizza endpoint VPC. Per ulteriori informazioni, consulta Chiavi disponibili per alcuni servizi nella Guida per l'utente di IAM.

Registrazione delle chiamate API del File di log Amazon CloudWatch in AWS CloudTrail

Amazon CloudWatch Logs è integrato con AWS CloudTrail, un servizio che offre un registro delle operazioni eseguite da un utente, da un ruolo o da un servizio AWS in CloudWatch Logs. CloudTrail acquisisce chiamate API effettuate da o per conto di un account AWS. Le chiamate acquisite includono le chiamate dalla console CloudWatch e le chiamate di codice alle operazioni delle API CloudWatch Logs. Se si crea un trail, è possibile abilitare la distribuzione continua di eventi CloudTrail in un bucket Amazon S3 includendo eventi per CloudWatch Logs. Se non configuri un percorso, puoi comunque visualizzare gli eventi più recenti nella console di CloudTrail in Cronologia eventi. Le informazioni raccolte da CloudTrail consentono di determinare la richiesta effettuata a CloudWatch Logs, l'indirizzo IP da cui è partita la richiesta, l'autore della richiesta, il momento in cui è stata eseguita e altri dettagli.

Per ulteriori informazioni su CloudTrail, incluso come configurarlo e abilitarlo, consulta la AWS CloudTrail Guida per l'utente.

Argomenti

Informazioni di CloudWatch Logs in CloudTrail

CloudTrail è abilitato sull'account AWS al momento della sua creazione. Quando si verifica un'attività evento supportata in CloudWatch Logs, tale attività viene registrata in un evento CloudTrail insieme ad altri eventi di servizio di AWS in Event history (Cronologia eventi). È possibile visualizzare, cercare e scaricare gli eventi recenti nell'account AWS. Per ulteriori informazioni, consulta Visualizzazione di eventi nella cronologia degli eventi di CloudTrail.

Per una registrazione continua degli eventi nell'account AWS, inclusi gli eventi per CloudWatch Logs, crea un percorso. Un percorso abilita la distribuzione da parte di CloudTrail dei file di log in un bucket Amazon S3. Per impostazione predefinita, quando si crea un trail nella console, il trail sarà valido in tutte le regioni AWS. Il trail registra gli eventi di tutte le Regioni nella partizione AWS e distribuisce i file di log nel bucket Amazon S3 specificato. Inoltre, è possibile configurare altri servizi AWS per analizzare con maggiore dettaglio e usare i dati evento raccolti nei log CloudTrail. Per ulteriori informazioni, consulta gli argomenti seguenti:

- Panoramica della creazione di un percorso
- Servizi e integrazioni CloudTrail supportati
- Configurazione delle notifiche Amazon SNS per CloudTrail
- Ricezione di file di log CloudTrail da più regioni e Ricezione di file di log CloudTrail da più account

CloudWatch Logs supporta la registrazione delle operazioni seguenti come eventi nei file di log CloudTrail:

- CancelExportTask
- CreateExportTask
- CreateLogGroup
- CreateLogStream
- DeleteDestination
- DeleteLogGroup
- DeleteLogStream
- DeleteMetricFilter
- DeleteRetentionPolicy
- DeleteSubscriptionFilter
- PutDestination
- PutDestinationPolicy
- PutMetricFilter
- PutResourcePolicy
- PutRetentionPolicy
- PutSubscriptionFilter
- StartQuery
- StopQuery
- TestMetricFilter

Per queste operazioni dell'API di CloudWatch Logs, in CloudTrail vengono registrati solo gli elementi di richiesta:

- DescribeDestinations

- DescribeExportTasks
- DescribeLogGroups
- DescribeLogStreams
- DescribeMetricFilters
- DescribeQueries
- DescribeResourcePolicies
- DescribeSubscriptionFilters
- FilterLogEvents
- GetLogEvents
- GetLogGroupFields
- GetLogRecord
- GetQueryResults

Ogni evento o voce di log contiene informazioni sull'utente che ha generato la richiesta. Le informazioni di identità consentono di determinare quanto segue:

- Se la richiesta è stata effettuata con le credenziali dell'utente IAM o root.
- Se la richiesta è stata effettuata con le credenziali di sicurezza temporanee per un ruolo o un utente federato.
- Se la richiesta è stata effettuata da un altro servizio AWS.

Per ulteriori informazioni, consulta Elemento userIdentity di CloudTrail.

Comprensione delle voci dei file di log di

Un trail è una configurazione che consente la distribuzione di eventi come i file di log in un bucket Amazon S3 specificato. I file di log di CloudTrail possono contenere una o più voci di log. Un evento rappresenta una singola richiesta da un'origine e include informazioni sull'operazione richiesta, sulla data e sull'ora dell'operazione, sui parametri richiesti e così via. I file di log CloudTrail non sono una traccia di pila ordinata delle chiamate API pubbliche e di conseguenza non devono apparire in base a un ordine specifico.

La voce di file di log seguente mostra che un utente ha chiamato l'operazione CloudWatch Logs CreateExportTask.

```
{
        "eventVersion": "1.03",
        "userIdentity": {
            "type": "IAMUser",
            "principalId": "EX_PRINCIPAL_ID",
            "arn": "arn:aws:iam::123456789012:user/someuser",
            "accountId": "123456789012",
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "userName": "someuser"
        },
        "eventTime": "2016-02-08T06:35:14Z",
        "eventSource": "logs.amazonaws.com",
        "eventName": "CreateExportTask",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "127.0.0.1",
        "userAgent": "aws-sdk-ruby2/2.0.0.rc4 ruby/1.9.3 x86_64-linux Seahorse/0.1.0",
        "requestParameters": {
            "destination": "yourdestination",
            "logGroupName": "yourloggroup",
            "to": 123456789012,
            "from": 0,
            "taskName": "yourtask"
        },
        "responseElements": {
            "taskId": "15e5e534-9548-44ab-a221-64d9d2b27b9b"
        },
        "requestID": "1cd74c1c-ce2e-12e6-99a9-8dbb26bd06c9",
        "eventID": "fd072859-bd7c-4865-9e76-8e364e89307c",
        "eventType": "AwsApiCall",
        "apiVersion": "20140328",
        "recipientAccountId": "123456789012"
}
```

Riferimento dell'agente CloudWatch Logs

> **Important**
>
> Questo riferimento riguarda l'agente di CloudWatch Logs precedente, reso obsoleto. Se utilizzi Instance Metadata Service versione 2 (IMDSv2), è necessario utilizzare il nuovo agente CloudWatch unificato. Se non stai utilizzando IMDSv2, consigliamo fortemente di usare il nuovo agente CloudWatch unificato invece del precedente agente dei registri. Per ulteriori informazioni sull'agente unificato più recente, consulta Raccolta di parametri e registri da istanze Amazon EC2 e da server On-Premise con l'agente di CloudWatch. Per informazioni sulla migrazione dall'agente di CloudWatch Logs precedente all'agente unificato, consulta Creazione del file di configurazione dell'agente CloudWatch tramite la procedura guidata.

L'agente CloudWatch Logs offre un modo automatizzato per inviare dati di registro a CloudWatch Logs da istanze Amazon EC2. L'agente include i componenti seguenti:

- Un plug-in per AWS CLI che esegue il push di dati di registro in CloudWatch Logs.
- Uno script (daemon) che avvia il processo per l'esecuzione del push di dati in CloudWatch Logs.
- Un processo cron che garantisce che il daemon sia sempre in esecuzione.

File di configurazione dell'agente

Il file di configurazione dell'agente CloudWatch Logs descrive le informazioni necessarie per l'agente CloudWatch Logs. La sezione [general] del file di configurazione dell'agente definisce le configurazioni comuni applicabili a tutti i flussi di log. La sezione [logstream] definisce le informazioni necessarie per l'invio di un file locale a un flusso di log in remoto. Puoi disporre di più sezioni [logstream], ma ciascuna deve avere un nome univoco all'interno del file di configurazione, ad esempio [logstream1], [logstream2] e così via. Il valore [logstream] e la prima riga di dati nel file di log definiscono l'identità del file di log.

```
[general]
state_file = value
logging_config_file = value
use_gzip_http_content_encoding = [true | false]
```

```
[logstream1]
log_group_name = value
log_stream_name = value
datetime_format = value
time_zone = [LOCAL|UTC]
file = value
file_fingerprint_lines = integer | integer-integer
multi_line_start_pattern = regex | {datetime_format}
initial_position = [start_of_file | end_of_file]
encoding = [ascii|utf_8|..]
buffer_duration = integer
batch_count = integer
batch_size = integer

[logstream2]
...
```

state_file

Specifica dove è archiviato il file di stato.

logging_config_file

(Opzionale) Specifica la posizione del file di configurazione di registrazione dell'agente. Se non specifichi in questa pagina un file di configurazione di registrazione dell'agente, verrà utilizzato il file predefinito awslogs.conf. La posizione del file di default è `/var/awslogs/etc/awslogs.conf` se hai installato l'agente con uno script oppure `/etc/awslogs/awslogs.conf` se hai installato l'agente con rpm. Il file è in formato di file di configurazione di Python (https://docs.python.org/2/library/logging.config.html#logging-config-fileformat). I logger con i nomi seguenti possono essere personalizzati.

```
cwlogs.push
cwlogs.push.reader
cwlogs.push.publisher
cwlogs.push.event
cwlogs.push.batch
cwlogs.push.stream
cwlogs.push.watcher
```

L'esempio seguente modifica il livello di lettore ed editore a WARNING, mentre il valore predefinito è INFO.

```
[loggers]
keys=root,cwlogs,reader,publisher

[handlers]
keys=consoleHandler

[formatters]
keys=simpleFormatter

[logger_root]
level=INFO
handlers=consoleHandler

[logger_cwlogs]
level=INFO
handlers=consoleHandler
qualname=cwlogs.push
propagate=0

[logger_reader]
level=WARNING
handlers=consoleHandler
qualname=cwlogs.push.reader
propagate=0

[logger_publisher]
level=WARNING
handlers=consoleHandler
qualname=cwlogs.push.publisher
propagate=0

[handler_consoleHandler]
class=logging.StreamHandler
level=INFO
formatter=simpleFormatter
args=(sys.stderr,)

[formatter_simpleFormatter]
format=%(asctime)s - %(name)s - %(levelname)s - %(process)d - %(threadName)s -
 %(message)s
```

use_gzip_http_content_encoding

Se impostato su VERO (di default), la codifica del contenuto gzip http è in grado di inviare payload compressi a CloudWatch Logs. Questo riduce l'utilizzo della CPU, NetworkOut e la latenza put. Per disabilitare questa funzionalità, aggiungi use_gzip_http_content_encoding = false alla sezione [general] del file di configurazione dell'agente di CloudWatch Logs e quindi riavvia l'agente.

> **Note**
>
> Questa impostazione è disponibile solo in awscli-cwlogs versione 1.3.3 e successive.

log_group_name

Specifica il gruppo di log di destinazione. Un gruppo di log viene creato automaticamente se non è già esistente. I nomi dei gruppi di log possono essere lunghi da 1 a 512 caratteri. I caratteri consentiti includono a-z, A-Z, 0-9, '_' (trattino basso), '-' (trattino), '/' (barra) e '.' (punto).

log_stream_name

Specifica il flusso di log di destinazione. Per definire un nome per il flusso di log, puoi utilizzare una stringa letterale o variabili predefinite ({instance_id}, {hostname} e {ip_address}) oppure una combinazione di entrambe. Un flusso di log viene creato automaticamente se non è già esistente.

datetime_format

Specifica il modo in cui il timestamp viene estratto dai log. Il timestamp viene utilizzato per recuperare eventi di log e generare di parametri. L'ora corrente viene utilizzata per ciascun evento di log se il valore di datetime_format non è fornito. Se il valore di datetime_format fornito non è valido per un determinato messaggio di log, verrà utilizzato il timestamp dell'ultimo evento di log con timestamp analizzato correttamente. Se non esistono eventi di log precedenti, viene utilizzata l'ora corrente.

I codici datetime_format comuni sono elencati di seguito. Puoi inoltre utilizzare qualsiasi codice datetime_format supportato da Python, datetime.strptime(). L'offset del fuso orario (%z) è supportato sebbene non sia supportato fino a Python 3.2, [+-]HHMM senza i due punti (:). Per ulteriori informazioni, consulta la pagina relativa al comportamento di strftime() e strptime().

%y: anno senza secolo come numero decimale a cui è aggiunto uno zero. 00, 01, ..., 99

%Y: anno con secolo come numero decimale. 1970, 1988, 2001, 2013

%b: mese come nome abbreviato nella lingua locale. Jan, Feb, ..., Dec (en_US);

c%B: mese come nome completo nella lingua locale. January, February, ..., December (en_US);

%m: mese come numero decimale a cui è aggiunto uno zero. 01, 02, ..., 12

%d: giorno del mese come numero decimale a cui è aggiunto uno zero. 01, 02, ..., 31

%H: ora (formato di 24 ore) come numero decimale a cui è aggiunto uno zero. 00, 01, ..., 23

%I: ora (formato di 12 ore) come numero decimale a cui è aggiunto uno zero. 01, 02, ..., 12

%p: equivalente locale di AM o PM.

%M: minuto come numero decimale a cui è aggiunto uno zero. 00, 01, ..., 59

%S: secondo come numero decimale a cui è aggiunto uno zero. 00, 01, ..., 59

%f: microsecondo come numero decimale, a cui è aggiunto uno zero a sinistra. 000000, ..., 999999

%z: differenza dall'ora UTC nel formato +HHMM o -HHMM. +0000, -0400, +1030

Formati di esempio:

```
Syslog: '%b %d %H:%M:%S', e.g. Jan 23 20:59:29

Log4j: '%d %b %Y %H:%M:%S', e.g. 24 Jan 2014 05:00:00

ISO8601: '%Y-%m-%dT%H:%M:%S%z', e.g. 2014-02-20T05:20:20+0000
```

time_zone

Specifica il fuso orario del timestamp dell'evento di log. I due valori supportati sono UTC e LOCAL. Il valore predefinito è LOCAL, utilizzato nel caso in cui il fuso orario non può essere dedotto sulla base di datetime_format.

file

Specifica i file di log di cui vuoi eseguire il push in CloudWatch Logs. File può puntare a un determinato file o più file (tramite caratteri jolly, come /var/log/system.log*). Viene eseguito il push in CloudWatch Logs solo del file più recente, in base all'ora di modifica del file. Ti consigliamo di utilizzare i caratteri jolly per specificare una serie di file dello stesso tipo, ad

esempio ccess_log.2014-06-01-01, access_log.2014-06-01-02 e così via, ma non file di più tipi, ad esempio access_log_80 e access_log_443. Per specificare più tipi di file, aggiungi un'altra voce di flusso di log al file di configurazione in modo che ogni tipo di file di log vada in un flusso di log distinto. I file compressi non sono supportati.

file_fingerprint_lines

Specifica l'intervallo di righe per identificare un file. I valori validi sono un numero o due numeri delimitati da trattino, ad esempio "1", "2-5". Il valore predefinito è "1" in modo da utilizzare la prima riga per calcolare l'impronta. Le righe dell'impronta non vengono inviate a CloudWatch Logs, a meno che non siano disponibili tutte le righe specificate.

multi_line_start_pattern

Specifica il modello per identificare l'inizio di un messaggio di log. Un messaggio di log è composto da una riga corrispondente al modello e da tutte le righe successive non corrispondenti al modello. I valori validi sono espressioni regolari o {datetime_format}. Quando utilizzi {datetime_format}, devi specificare l'opzione datetime_format. Il valore predefinito è "^[^\s]", in modo tale che le righe che iniziano con caratteri diversi da spazi vuoti chiudono il messaggio di log precedente e iniziano un nuovo messaggio di log.

initial_position

Specifica il punto da cui iniziare a leggere i dati (start_of_file o end_of_file). Il valore predefinito è start_of_file. È utilizzato solo se non vi è uno stato reso persistente per tale flusso di log.

encoding

Specifica la codifica del file di log in modo che il file può essere letto correttamente. Il valore predefinito è utf_8. Possono qui essere utilizzate le codifiche supportate da Python codecs.decode().

> **Warning**
>
> La specificazione di una codifica non corretta potrebbe causare una perdita di dati, in quanto i caratteri che non possono essere decodificati saranno sostituiti da altri caratteri.

Sono elencate di seguito alcune codifiche comuni:

```
ascii, big5, big5hkscs, cp037, cp424, cp437, cp500, cp720, cp737,
cp775, cp850, cp852, cp855, cp856, cp857, cp858, cp860, cp861, cp862,
```

```
cp863, cp864, cp865, cp866, cp869, cp874, cp875, cp932, cp949, cp950,
cp1006, cp1026, cp1140, cp1250, cp1251, cp1252, cp1253, cp1254, cp1255,
cp1256, cp1257, cp1258, euc_jp, euc_jis_2004, euc_jisx0213, euc_kr,
gb2312, gbk, gb18030, hz, iso2022_jp, iso2022_jp_1, iso2022_jp_2,
iso2022_jp_2004, iso2022_jp_3, iso2022_jp_ext, iso2022_kr, latin_1,
iso8859_2, iso8859_3, iso8859_4, iso8859_5, iso8859_6, iso8859_7,
iso8859_8, iso8859_9, iso8859_10, iso8859_13, iso8859_14, iso8859_15,
iso8859_16, johab, koi8_r, koi8_u, mac_cyrillic, mac_greek, mac_iceland,
mac_latin2, mac_roman, mac_turkish, ptcp154, shift_jis, shift_jis_2004,
shift_jisx0213, utf_32, utf_32_be, utf_32_le, utf_16, utf_16_be,
utf_16_le, utf_7, utf_8, utf_8_sig
```

buffer_duration

Specifica la durata del raggruppamento di eventi di log. Il valore minimo è 5.000 ms e il valore predefinito è 5.000 ms.

batch_count

Specifica il numero massimo di eventi di log in un batch, fino a un massimo di 10.000. Il valore predefinito è 10000.

batch_size

Specifica la dimensione massima di eventi di log in un batch in byte, fino a un massimo di 1.048.576 byte. Il valore predefinito è 1048576 byte. Questa dimensione viene calcolata come la somma di tutti i messaggi di eventi in UTF-8, più 26 byte per ogni evento di log.

Utilizzo dell'agente di CloudWatch Logs con proxy HTTP

Puoi utilizzare l'agente di CloudWatch Logs con proxy HTTP.

> **Note**
>
> I proxy HTTP sono supportati in awslogs-agent-setup.py versione 1.3.8 o successive.

Per utilizzare l'agente di CloudWatch Logs con proxy HTTP

1. Completa una delle seguenti operazioni:

a. Per una nuova installazione dell'agente di CloudWatch Logs, esegui i comandi seguenti:

```
curl https://s3.amazonaws.com/aws-cloudwatch/downloads/latest/awslogs-agent-setup.py -O
```

```
sudo python awslogs-agent-setup.py --region us-east-1 --http-proxy http://your/proxy --https-proxy http://your/proxy --no-proxy 169.254.169.254
```

Per mantenere l'accesso al servizio di metadati Amazon EC2 in istanze EC2, utilizza --no-proxy 169.254.169.254 (consigliato). Per ulteriori informazioni, consulta Metadati dell'istanza e dati dell'utente nella Guida per l'utente di Amazon EC2 per le istanze Linux.

Nei valori per `http-proxy` e `https-proxy`, è necessario specificare l'URL nella sua interezza.

b. Per un'installazione esistente dell'agente di CloudWatch Logs, modifica /var/awslogs/etc/proxy.conf e quindi aggiungi i proxy:

```
HTTP_PROXY=
HTTPS_PROXY=
NO_PROXY=
```

2. Riavvia l'agente per rendere effettive le modifiche:

```
sudo service awslogs restart
```

Se utilizzi Amazon Linux 2, utilizza il comando seguente per riavviare l'agente:

```
sudo service awslogsd restart
```

Compartimentazione dei file di configurazione dell'agente di CloudWatch Logs

Se utilizzi awslogs-agent-setup.py versione 1.3.8 o successive con awscli-cwlogs 1.3.3 o versioni successive, puoi importare diverse configurazioni del flusso per vari componenti in modo indipendente, creando ulteriori file di configurazione nella directory /var/awslogs/etc/config/. Quando l'agente di CloudWatch Logs viene avviato, include tutte le configurazioni dei flussi in questi file

di configurazione aggiuntivi. Le proprietà di configurazione nella sezione [general] devono essere definite nel file di configurazione principale (/var/awslogs/etc/awslogs.conf) e vengono ignorate in tutti i file di configurazione aggiuntivi in /var/awslogs/etc/config/.

Se non disponi di una directory /var/awslogs/etc/config/ perché hai installato l'agente con rpm, puoi in alternativa utilizzare la directory /etc/awslogs/config/.

Riavvia l'agente per rendere effettive le modifiche:

```
sudo service awslogs restart
```

Se utilizzi Amazon Linux 2, utilizza il comando seguente per riavviare l'agente:

```
sudo service awslogsd restart
```

Domande frequenti sull'agente di CloudWatch Logs

Quali tipi di rotazione di file sono supportati?

Sono supportati i seguenti meccanismi di rotazione di file:

- Ridenominazione di file di log esistenti con un suffisso numerico, quindi nuova creazione del file di log vuoto originale. Ad esempio, /var/log/syslog.log viene rinominato in /var/log/syslog.log.1. Se /var/log/syslog.log.1 esiste già da una rotazione precedente, viene rinominato in /var/log/syslog.log.2.
- Troncamento del file di log originale in vigore dopo la creazione di una copia. Ad esempio, /var/log/syslog.log viene copiato in /var/log/syslog.log.1 e /var/log/syslog.log viene troncato. In questo caso potrebbe verificarsi una perdita di dati, quindi fai attenzione con l'uso di questo meccanismo di rotazione di file.
- Creazione di un nuovo file con un modello comune come il precedente. Ad esempio, /var/log/syslog.log.2014-01-01 rimane e viene creato /var/log/syslog.log.2014-01-02.

L'impronta (ID origine) del file viene calcolata sottoponendo a hashing la chiave di flusso di log e la prima riga del contenuto del file. Per sovrascrivere questo comportamento, puoi utilizzare l'opzione file_fingerprint_lines. Quando si verifica la rotazione del file, il nuovo file deve avere un nuovo contenuto e il vecchio file non deve avere contenuto aggiuntivo; l'agente invia il nuovo file dopo aver completato la lettura del vecchio file.

Come posso determinare quale versione di agente sto utilizzando?

Se hai utilizzato uno script di installazione per installare l'agente di CloudWatch Logs, puoi utilizzare /var/awslogs/bin/awslogs-version.sh per determinare quale versione dell'agente stai utilizzando. Verrà stampata la versione dell'agente e le sue dipendenze principali. Se hai utilizzato yum per installare l'agente di CloudWatch Logs, puoi utilizzare "yum info awslogs" e "yum info aws-cli-plugin-cloudwatch-logs" per controllare la versione dell'agente di CloudWatch Logs e del plug-in.

Come vengono convertite le voci di log in eventi di log?

Gli eventi di log contengono due proprietà: il timestamp del momento in cui si è verificato l'evento e il messaggio di log non elaborato. Per impostazione predefinita, le righe che iniziano con caratteri diversi da spazi vuoti chiudono il messaggio di log precedente se ne esiste uno e iniziano un nuovo messaggio di log. Per sovrascrivere questo comportamento, puoi utilizzare multi_line_start_pattern e le righe che corrispondono al modello iniziano un nuovo messaggio di log. Il modello può essere qualsiasi espressione regolare o "{datetime_format}". Ad esempio, se la prima riga di ogni messaggio di log contiene un timestamp come "2014-01-02T13:13:01Z", allora multi_line_start_pattern può essere impostato su "\d{4}-\d{2}-\d{2}T\d{2}:\d{2}:\d{2}Z". Per semplificare la configurazione, puoi utilizzare la variabile "{datetime_format}" se hai specificato l'opzione datetime_format. Per lo stesso esempio, se datetime_format è impostato su "%Y-%m-%dT%H:%M:%S%z", multi_line_start_pattern può essere semplicemente "{datetime_format}".

L'ora corrente viene utilizzata per ciascun evento di log se il valore di datetime_format non è fornito. Se il valore di datetime_format fornito non è valido per un determinato messaggio di log, verrà utilizzato il timestamp dell'ultimo evento di log con timestamp analizzato correttamente. Se non esistono eventi di log precedenti, viene utilizzata l'ora corrente. Viene registrato un messaggio di avviso quando un evento di log utilizza l'ora corrente o l'ora dell'evento di log precedente.

I timestamp vengono utilizzati per recuperare eventi di log e generare parametri, perciò se specifichi il formato errato, gli eventi di log potrebbero diventare non recuperabili e generare i parametri errati.

Come vengono raggruppatigli eventi di log?

Un batch diviene completo e viene pubblicato quando si verifica qualsiasi delle condizioni seguenti:

1. La quantità di tempo di buffer_duration è trascorsa a partire dall'aggiunta del primo evento di log.

2. È stato accumulato un valore inferiore di batch_size log di eventi di log, ma l'aggiunta del nuovo evento di log supera il valore di batch_size.
3. Il numero di eventi di log ha raggiunto il valore di batch_count.
4. Gli eventi di log dal batch non si estendono per più di 24 ore, ma l'aggiunta del nuovo evento di log supera il vincolo di 24 ore.

Cosa può causare l'omissione o il troncamento di voci di log, eventi di log o batch?

Per seguire il vincolo dell'operazione `PutLogEvents`, i seguenti problemi potrebbero causare l'omissione di un evento di log o di un batch.

> **Note**
>
> L'agente di CloudWatch Logs scrive un avviso nel proprio registro quando i dati vengono omessi.

1. Se la dimensione di un evento di log supera 256 KB, l'evento di log viene completamente ignorato.
2. Se il timestamp dell'evento di log è superiore a 2 ore successive all'ora attuale, l'evento di log viene ignorato.
3. Se il timestamp dell'evento di log è superiore a 14 giorni antecedenti il giorno attuale, l'evento di log viene ignorato.
4. Se qualsiasi evento di log è precedente al periodo di conservazione del gruppo di log, l'intero batch viene ignorato.
5. Se il batch di eventi di log in una singola richiesta `PutLogEvents` si estende per più di 24 ore, l'operazione `PutLogEvents` ha esito negativo.

L'arresto dell'agente causa perdita di dati o duplicati?

No, purché il file di stato sia disponibile e non si sia verificata alcuna rotazione del file a partire dall'ultima esecuzione. L'agente di CloudWatch Logs può avviarsi dal punto in cui si è arrestato e continuare a eseguire il push dei dati di registro.

Posso indirizzare diversi file di log dallo stesso host o da host diversi allo stesso flusso di log?

La configurazione di più origini di log per l'invio di dati a un unico flusso di log non è supportata.

Quali chiamate API effettua l'agente (o quali operazioni devo aggiungere alla mia policy IAM)?

L'agente CloudWatch Logs richiede le operazioni `CreateLogGroup`, `CreateLogStream`, `DescribeLogStreams`, e `PutLogEvents`. Se utilizzi l'ultimo agente, `DescribeLogStreams` non è necessario. Consulta la policy IAM di esempio riportata di seguito.

```
{
"Version": "2012-10-17",
"Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "logs:CreateLogGroup",
      "logs:CreateLogStream",
      "logs:PutLogEvents",
      "logs:DescribeLogStreams"
    ],
    "Resource": [
      "arn:aws:logs:*:*:*"
    ]
  }
 ]
}
```

Non voglio che l'agente di CloudWatch Logs crei gruppi di registro o flussi di registro in modo automatico. Come posso impedire all'agente di ricreare gruppi di log e flussi di log?

Nella tua policy IAM, puoi limitare l'agente alle sole operazioni seguenti: `DescribeLogStreams`, `PutLogEvents`.

Prima di revocare le autorizzazioni `CreateLogStream` e `CreateLogGroup` all'agente, assicurati di creare sia i gruppi di log che i flussi di log che devono essere utilizzati dall'agente. L'agente di log non può creare flussi di log in un gruppo di log creato a meno che non disponga delle autorizzazioni `CreateLogStream` e `CreateLogGroup`.

Quali log dovrei esaminare durante la risoluzione dei problemi?

Il log di installazione dell'agente si trova in `/var/log/awslogs-agent-setup.log` e il log dell'agente si trova in `/var/log/awslogs.log`.

Monitoraggio dell'utilizzo con i parametri di CloudWatch

CloudWatch Logs invia i parametri ad Amazon CloudWatch ogni minuto.

Parametri di CloudWatch Logs

Il namespace `AWS/Logs` include i parametri descritti di seguito.

Parametro	Descrizione
`CallCount`	Il numero di operazioni API specificate eseguite nel tuo account. `CallCount` è un parametro di utilizzo del servizio CloudWatch Logs. Per ulteriori informazioni, consulta Parametri di utilizzo del servizio CloudWatch Logs. Dimensioni valide: Classe, Risorsa, Servizio, Tipo Statistiche valide: Sum Unità: nessuna
`DeliveryErrors`	Il numero di log eventi per i quali CloudWatch Logs ha ricevuto un errore durante l'inoltro dei dati alla destinazione di sottoscrizione. Se il servizio di destinazione restituisce un errore non irreversibile, ad esempio un'eccezione di limitazione o un'eccezione di servizio non irreversibile (HTTP 5xx ad esempio), CloudWatch Logs continua a ripetere la consegna per un massimo di 24 ore. CloudWatch Logs non prova a riconsegnare se l'errore è un errore irreversibile, ad esempio `AccessDeniedException` o `ResourceNotFoundException`. Dimensioni valide: LogGroupName, DestinationType, FilterName Statistiche valide: Sum Unità: nessuna
`DeliveryThrottling`	Il numero di log eventi per i quali CloudWatch Logs è stato limitato durante l'inoltro dei dati alla destinazione di sottoscrizione.

Parametro	Descrizione
	Se il servizio di destinazione restituisce un errore non irreversibile, ad esempio un'eccezione di limitazione o un'eccezione di servizio non irreversibile (HTTP 5xx ad esempio), CloudWatch Logs continua a ripetere la consegna per un massimo di 24 ore. CloudWatch Logs non prova a riconsegnare se l'errore è un errore irreversibile, ad esempio `AccessDeniedException` o `ResourceNotFoundException`. Dimensioni valide: LogGroupName, DestinationType, FilterName Statistiche valide: Sum Unità: nessuna
`EMFParsingErrors`	Il numero di errori di analisi riscontrati durante l'elaborazione dei log in formato dei parametri incorporato. Tali errori si verificano quando i registri vengono identificati come formato dei parametri incorporato ma non seguono il formato corretto. Per ulteriori informazioni sul formato dei parametri incorporato, consulta la sezione Specification: Embedded metric format (Specifica: formato dei parametri incorporato). Dimensioni valide: `LogGroupName` Statistiche valide: Sum Unità: nessuna

Parametro	Descrizione
`EMFValidationErrors`	Il numero di errori di convalida riscontrati durante l'elaborazione dei log in formato dei parametri incorporato. Questi errori si verificano quando le definizioni dei parametri nei log in formato dei parametri incorporato non aderiscono al formato dei parametri incorporato e alle specifiche di `MetricDatum` . Per ulteriori informazioni sul formato dei parametri incorporato, consulta Specifiche: formato dei parametri incorporati. Per informazioni sul tipo di dati `MetricDatum` , consulta MetricDatum in Informazioni di riferimento sulle API Amazon CloudWatch. **Note** Alcuni errori di convalida possono causare la mancata pubblicazione di più metriche all'interno di un log EMF. Ad esempio, tutti i parametri impostati con un namespace non valido verranno eliminati. Dimensioni valide: `LogGroupName` Statistiche valide: Sum Unità: nessuna
`ErrorCount`	Il numero di operazioni API eseguite nel tuo account che sono risultate in errori. `ErrorCount` è un parametro di utilizzo del servizio CloudWatch Logs. Per ulteriori informazioni, consulta Parametri di utilizzo del servizio CloudWatch Logs. Dimensioni valide: Classe, Risorsa, Servizio, Tipo Statistiche valide: Sum Unità: nessuna

Parametro	Descrizione
`ForwardedBytes`	Il volume dei log eventi in byte compressi inoltrati alla destinazione di sottoscrizione. Dimensioni valide: LogGroupName, DestinationType, FilterName Statistiche valide: Sum Unità: byte
`ForwardedLogEvents`	Il numero dei log eventi inoltrati alla destinazione di sottoscrizione. Dimensioni valide: LogGroupName, DestinationType, FilterName Statistiche valide: Sum Unità: nessuna
`IncomingBytes`	Il volume dei log eventi in byte non compressi caricati in CloudWatch Logs. Se utilizzato con la dimensione `LogGroupName` , corrisponde al volume degli eventi di log in byte non compressi caricati nel gruppo di log. Dimensioni valide: LogGroupName Statistiche valide: Sum Unità: byte
`IncomingLogEvents`	Il numero di log eventi caricati in CloudWatch Logs. Se utilizzato con la dimensione `LogGroupName` , corrisponde al numero di eventi di log caricati nel gruppo di log. Dimensioni valide: LogGroupName Statistiche valide: Sum Unità: nessuna

Parametro	Descrizione
`LogEvents WithFindings`	Il numero di log eventi corrispondenti a una stringa di dati in corso di verifica tramite la funzione di protezione dei dati di CloudWatch Logs. Per ulteriori informazioni, consulta Incremento della protezione dei dati di log sensibili con il mascheramento. Dimensioni valide: nessuna Statistiche valide: Sum Unità: nessuna
`ThrottleCount`	Il numero di operazioni API eseguite nell'account che sono state limitate a causa delle quote di utilizzo. `ThrottleCount` è un parametro di utilizzo del servizio CloudWatch Logs. Per ulteriori informazioni, consulta Parametri di utilizzo del servizio CloudWatch Logs. Dimensioni valide: Classe, Risorsa, Servizio, Tipo Statistiche valide: Sum Unità: nessuna

Dimensioni per i parametri di CloudWatch Logs

Di seguito sono elencate le dimensioni che è possibile utilizzare con i parametri di CloudWatch Logs.

Dimensione	Descrizione
`LogGroupName`	Il nome del gruppo di log di CloudWatch Logs per il quale visualizzare i parametri.
`DestinationType`	Destinazione della sottoscrizione per i dati CloudWatch Logs, che può essere AWS Lambda, Amazon Kinesis Data Streams o Amazon Kinesis Data Firehose.

Dimensione	Descrizione
`FilterName`	Nome del filtro di sottoscrizione che sta inviando dati dal gruppo di log alla destinazione. Il nome del filtro di sottoscrizione viene automaticamente convertito da CloudWatch in ASCII e qualsiasi carattere non supportato viene sostituito con un punto interrogativo (?).

Parametri di utilizzo del servizio CloudWatch Logs

CloudWatch Logs invia parametri a CloudWatch che tengono traccia l'utilizzo delle operazioni API di CloudWatch Logs. Questi parametri corrispondono alle quote di servizio AWS. Il monitoraggio di questi parametri consente di gestire in modo proattivo le tue quote. Per ulteriori informazioni, consulta Service Quotas Integration and Usage Metrics (Integrazione di quote di servizio e parametri di utilizzo).

Ad esempio, è possibile tenere traccia dei parametri `ThrottleCount` o imposta un allarme per tale parametro. Se il valore di questo parametro aumenta, è consigliabile richiedere un aumento di quota per l'operazione API che viene limitata. Per ulteriori informazioni sulle quote di servizio CloudWatch Logs, consulta CloudWatch Registra le quote.

CloudWatch Logs pubblica i parametri di utilizzo delle quote del servizio ogni minuto sia in `AWS/Usage` e `AWS/Logs` spazio dei nomi.

La tabella seguente elenca i parametri di utilizzo del servizio pubblicati da CloudWatch Logs. Questi parametri non hanno un'unità specificata. La statistica più utile per questi parametri è `SUM`, che rappresenta il conteggio totale delle operazioni per il periodo di 1 minuto.

Ciascuno di questi parametri vengono pubblicati con i valori per tutte le dimensioni di `Service`, `Class`, `Type`, e `Resource`. Vengono inoltre pubblicati con una singola dimensione chiamata `Account Metrics`. Utilizzare la dimensione `Account Metrics` per visualizzare la somma dei parametri per tutte le operazioni API nel tuo account. Utilizzare le altre dimensioni e specificare il nome di un'operazione API per la dimensione `Resource` per trovare i parametri per quella particolare API.

Parametri

Parametro	Descrizione
`CallCount`	Il numero di operazioni specificate eseguite nel tuo account. `CallCount` è pubblicato in entrambi spazi dei nomi `AWS/Usage` e `AWS/Logs`.
`ErrorCount`	Il numero di operazioni API eseguite nel tuo account che sono risultate in errori. `ErrorCount` è pubblicato solo nel `AWS/Logs`.
`ThrottleCount`	Il numero di operazioni API eseguite nell'account che sono state limitate a causa delle quote di utilizzo. `ThrottleCount` è pubblicato solo nel `AWS/Logs`.

Dimensioni

Dimensione	Descrizione
`Account metrics`	Utilizzare questa dimensione per ottenere una somma dei parametri in tutte le API di CloudWatch Logs. Se si desidera visualizzare i parametri per una determinata API, utilizzar e le altre dimensioni elencate in questa tabella e specificare il nome API come valore di `Resource`.
`Service`	Il nome del servizio AWS contenente la risorsa. Per i parametri di utilizzo di CloudWatch Logs, il valore per questa dimensione è `Logs`.
`Class`	La classe della risorsa monitorata. I parametri di utilizzo dell'API di CloudWatch Logs utilizzano questa dimensione con un valore di `None`.
`Type`	Il tipo di risorsa monitorata. Attualmente, quando la dimensione `Service` è `Logs`, l'unico valore valido per `Type` è `API`.

Dimensione	Descrizione
`Resource`	Il nome dell'operazione API. I valori validi includono tutti i nomi delle operazioni API elencati in Actions (Operazioni). Ad esempio, `PutLogEvents`

CloudWatch Registra le quote

Le tabelle seguenti forniscono le quote di servizio predefinite, note anche come limiti, per i CloudWatch registri di un account. AWS La maggior parte di queste quote di servizio, ma non tutte, sono elencate nello spazio dei nomi Amazon CloudWatch Logs nella console Service Quotas. Per richiedere un aumento delle quote per tali quote, consultare la procedura più avanti in questa sezione.

Risorsa	Quota predefinita
Dimensione batch	La dimensione massima di un batch è di 1.048.576 byte. Questa dimensione viene calcolata come la somma di tutti i messaggi di eventi in UTF-8, più 26 byte per ogni log eventi. Questa quota non può essere modificata.
Archiviazione di dati	Fino a 5 GB di spazio di archiviazione di dati gratis. Questa quota non può essere modificata.
CreateLogGroup	5 transazioni al secondo (TPS/account/Regione), dopodiché le transazioni vengono limitate. È possibile richiedere un aumento della quota.
CreateLogStream	50 transazioni al secondo (TPS/account/regione), dopodiché le transazioni vengono limitate. È possibile richiedere un aumento della quota.
DeleteLogGroup	5 transazioni al secondo (TPS/account/Regione), dopodiché le transazioni vengono limitate. È possibile richiedere un aumento della quota.
DeleteLogStream	5 transazioni al secondo (TPS/account/Regione), dopodiché le transazioni vengono limitate. È possibile richiedere un aumento della quota.
DescribeLogGroups	5 transazioni al secondo (TPS per account/per regione). È possibile richiedere un aumento della quota.

Risorsa	Quota predefinita
DescribeLogStreams	5 transazioni al secondo (TPS per account/per regione). È possibile richiedere un aumento della quota.
Campi di log rilevati	CloudWatch Logs Insights è in grado di rilevare un massimo di 1000 campi di eventi di registro in un gruppo di log. Questa quota non può essere modificata. Per ulteriori informazioni, consulta Registri supportati e campi rilevati.
Campi di log estratti nei log JSON	CloudWatch Logs Insights può estrarre un massimo di 200 campi di eventi di registro da un registro JSON. Questa quota non può essere modificata. Per ulteriori informazioni, consulta Registri supportati e campi rilevati.
Attività di esportazione	Un'attività di esportazione (in esecuzione o in attesa) attiva alla volta, per ogni account. Questa quota non può essere modificata.

Risorsa	Quota predefinita
FilterLogEvents	25 richieste al secondo negli Stati Uniti orientali (Virginia settentrionale). 10 richieste al secondo nelle seguenti regioni: • Stati Uniti orientali (Ohio) • Stati Uniti occidentali (California settentrionale) • Stati Uniti occidentali (Oregon) • Africa (Città del Capo) • Asia Pacific (Hong Kong) • Asia Pacifico (Mumbai) • Asia Pacifico (Seoul) • Asia Pacifico (Singapore) • Asia Pacifico (Tokyo) • Asia Pacifico (Sydney) • Canada (Central) • Europa (Irlanda) • Europa (Londra) • Europa (Milano) • Europa (Parigi) • Europa (Stoccolma) • Medio Oriente (Bahrein) • Sud America (San Paolo) • AWS GovCloud (Stati Uniti orientali) • AWS GovCloud (Stati Uniti occidentali) 5 richieste al secondo in tutte le altre regioni. Questa quota non può essere modificata.

Risorsa	Quota predefinita
GetLogEvents	30 richieste al secondo in Europa (Parigi). 25 richieste al secondo nelle seguenti regioni: • Stati Uniti orientali (Virginia settentrionale) • Stati Uniti orientali (Ohio) • Stati Uniti occidentali (California settentrionale) • Africa (Città del Capo) • Asia Pacific (Hong Kong) • Asia Pacifico (Mumbai) • Asia Pacifico (Seoul) • Asia Pacifico (Singapore) • Asia Pacifico (Tokyo) • Asia Pacifico (Sydney) • Canada (Centrale) • Europa (Londra) • Europa (Milano) • Europa (Stoccolma) • Medio Oriente (Bahrein) • Sud America (San Paolo) • AWS GovCloud (Stati Uniti orientali) • AWS GovCloud (Stati Uniti occidentali) 10 richieste al secondo in tutte le altre regioni. Questa quota non può essere modificata. In caso di costante elaborazione di nuovi dati, è consiglia bile sottoscrivere un abbonamento. In caso di necessità dei dati storici, è consigliabile esportare i dati in Amazon S3.

Risorsa	Quota predefinita
Dati in ingresso	Fino a 5 GB di dati in ingresso gratis. Questa quota non può essere modificata.
Sessioni simultanee di Live Tail.	15 sessioni simultanee. È possibile richiedere un aumento della quota.
Live Tail: gruppi di log cercati in una sessione.	Massimo 10 gruppi di log scansionati in una sessione Live Tail. Questa quota non può essere modificata.
Dimensione del log eventi	256 KB (massimo). Questa quota non può essere modificata.
Gruppi di log	1.000.000 gruppi di log per account e per Regione. È possibile richiedere un aumento della quota. Non vi è alcuna quota per il numero di flussi di log che possono appartenere a un gruppo di log.
Filtri di parametri	100 per gruppo di log. Questa quota non può essere modificata.
Parametri del formato dei parametri incorporato	100 parametri per log eventi e 30 dimensioni per parametro. Per ulteriori informazioni sul formato metrico incorporato, consulta Specificazione: Embedded Metric Format nella Amazon CloudWatch User Guide.
PutLogEvents	La dimensione massima del batch di una PutLogEvents richiesta è di 1 MB. 800 transazioni al secondo per account per Regione, ad eccezione delle seguenti Regioni in cui la quota è di 1.500 transazioni al secondo per account per Regione: Stati Uniti orientali (Virginia settentrionale), Stati Uniti occidentali (Oregon) e Europa (Irlanda). È possibile richiedere un aumento della quota di limitazione al secondo utilizzando il servizio. Service Quotas

Risorsa	Quota predefinita
Timeout di esecuzione della query	Le query in CloudWatch Logs Insights scadono dopo 60 minuti. Questo limite di tempo non può essere modificato.
Gruppi di log di query	È possibile interrogare un massimo di 50 gruppi di log in una singola CloudWatch query di Logs Insights. Questa quota non può essere modificata.
Query simultanee	Un massimo di 30 interrogazioni simultanee di CloudWatch Logs Insights, incluse le query che sono state aggiunte ai dashboard. Questa quota non può essere modificata.
Richiedi disponibilità	Le query create nella console sono disponibili per 30 giorni, tramite il comando Cronologia. Questo periodo di disponibilità non può essere modificato. Le definizioni delle query create utilizzando non scadono. PutQueryDefinition
Disponibilità dei risultati delle query	I risultati di una query sono disponibili per 7 giorni. Questo periodo di disponibilità non può essere modificato.
Risultati della query visualizzati nella console	Per impostazione predefinita, sulla console vengono visualizzate fino a 1.000 righe di risultati di query. È possibile utilizzare il comando **`limit`** in una query per aumentarlo fino a 10.000 righe. Per ulteriori informazioni, consulta CloudWatch Sintassi delle query di Logs Insights.
Espressioni regolari	Fino a 5 modelli di filtro contenenti espressioni regolari per ogni gruppo di log durante la creazione di filtri di parametri o filtri di sottoscrizione. Questa quota non può essere modificata. Fino a 2 espressioni regolari per ogni modello di filtro durante la creazione di un modello di filtro delimitato o JSON per filtri di parametri e filtri di sottoscrizione o quando si filtrano log eventi.

Risorsa	Quota predefinita
Policy delle risorse	Fino a 10 politiche relative alle risorse di CloudWatch registro per regione per account. Questa quota non può essere modificata.
Query salvate	Puoi salvare fino a 1000 query di CloudWatch Logs Insights, per regione per account. Questa quota non può essere modificata.
Filtri di sottoscrizione	2 per gruppo di log. Questa quota non può essere modificata.

Gestione delle quote del servizio CloudWatch Logs

CloudWatch Logs si è integrato con Service Quotas, AWS un servizio che consente di visualizzare e gestire le quote da una posizione centrale. Per ulteriori informazioni, consulta Cos'è Service Quotas? nella Guida per l'utente di Service Quotas.

Service Quotas semplifica la ricerca del valore delle quote del servizio CloudWatch Logs.

AWS Management Console

Per visualizzare le quote del servizio CloudWatch Logs utilizzando la console

1. Apri la console Service Quotas all'indirizzo https://console.aws.amazon.com/servicequotas/
2. Nel pannello di navigazione, scegli ServiziAWS .
3. Dall'elenco dei AWS servizi, cerca e seleziona Amazon CloudWatch Logs.

   Nell'elenco Service Quotas, è possibile visualizzare il nome della quota di servizio, il valore applicato (se è disponibile), la quota predefinita AWS e se il valore della quota è adattabile.
4. Per visualizzare ulteriori informazioni su una quota di servizio, ad esempio la descrizione, scegli il nome della quota.
5. (Facoltativo) Per richiedere un aumento della quota, selezionare la quota che si desidera aumentare, selezionare Richiedi aumento quota, immettere o selezionare le informazioni richieste e selezionare Richiedi.

Per maggiori informazioni sulle quote di servizio utilizzando la console, consulta Guida per l'utente di Service Quotas. Per richiedere un aumento delle quote, consulta Richiesta di aumento delle quote nella Guida per l'utente di Service Quotas.

AWS CLI

Per visualizzare le quote del servizio CloudWatch Logs, utilizza AWS CLI

Eseguire il comando seguente per visualizzare le quote di CloudWatch registro predefinite.

```
aws service-quotas list-aws-default-service-quotas \
    --query 'Quotas[*].
{Adjustable:Adjustable,Name:QuotaName,Value:Value,Code:QuotaCode}' \
    --service-code logs \
    --output table
```

Per utilizzare meglio le quote di servizio utilizzando il AWS CLI, vedere Service Quotas Command AWS CLI Reference. Per richiedere un aumento delle quote, consultare il comando `request-service-quota-increase` nella Documentazione di riferimento sui comandiAWS CLI.

Cronologia dei documenti

La tabella seguente descrive le modifiche importanti in ogni versione della CloudWatch Logs User Guide, a partire da giugno 2018. Per ricevere notifiche sugli aggiornamenti di questa documentazione, puoi abbonarti a un feed RSS.

Modifica	Descrizione	Data
CloudWatch Logs aggiunge il supporto alla sintassi del pattern di filtro delle espressioni regolari per Live Tail	Ora puoi personalizzare ulteriormente le operazioni di ricerca e corrispondenza per soddisfare le tue esigenze con espressioni regolari flessibili all'interno dei modelli di filtro Live Tail. Per ulteriori informazioni, consulta la sintassi del pattern di filtro nella Amazon CloudWatch Logs User Guide.	13 novembre 2023
CloudWatch Logs aggiunge la sintassi del pattern di filtro delle espressioni regolari, il supporto per filtri metrici, filtri di sottoscrizione ed eventi di log dei filtri.	Ora puoi personalizzare ulteriormente le operazioni di ricerca e corrispondenza per soddisfare le tue esigenze con espressioni regolari flessibili all'interno di modelli di filtro. Per ulteriori informazioni, consulta la sintassi del pattern di filtro nella Amazon CloudWatch Logs User Guide.	5 settembre 2023
CloudWatch Logs Insights aggiunge un comando pattern	Ora puoi utilizzare pattern nelle tue query di CloudWatch Logs Insights per raggruppare automaticamente i dati di registro in modelli. Un pattern è una struttura di testo condivisa che ricorre tra i	17 luglio 2023

	campi dei log. Per ulteriori informazioni, consulta pattern nella Amazon CloudWatch Logs User Guide.	
CloudWatch Logs Insights aggiunge un comando dedup	È ora possibile utilizzare dedup nelle query di CloudWatch Logs Insights per rimuovere i risultati duplicati in base a valori specifici nei campi specificati. Per ulteriori informazioni, consulta dedup nella Amazon CloudWatch Logs User Guide.	20 giugno 2023
Policy di protezione dei dati a livello di account	Ora puoi impostare le policy di protezione dei dati a livello di account. Queste policy a livello di account possono verificare e mascherare le informazioni riservate dei log eventi in tutti i gruppi di log dell'account. Per ulteriori informazioni, consulta Aiutare a proteggere i dati di log sensibili con il mascheramento nella Amazon CloudWatch Logs User Guide.	8 giugno 2023

Funzionalità Live Tail aggiunta	CloudWatch I log hanno aggiunto la funzionalità Live Tail, quindi puoi scansionare i log man mano che vengono inseriti per facilitare la risoluzione dei problemi. Facoltativamente, puoi filtrare il flusso di log eventi visualizzato in base a termini specificati ed evidenziare anche i log eventi con termini specifici. Per ulteriori informazioni, consulta Use live tail to view logs in near real time.	6 giugno 2023
CloudWatchLogsReadOnlyAccesspolitica aggiornata	CloudWatch Registra le autorizzazioni aggiunte a. CloudWatchLogsReadOnlyAccess Le `logs:StopLiveTail` autorizzazioni `logs:StartLiveTail` e sono state aggiunte in modo che gli utenti con questo criterio possano utilizzare la console per avviare e interrompere le sessioni live tail di CloudWatch Logs. Per ulteriori informazioni, consulta Use live tail to view logs in near real time.	6 giugno 2023

CloudWatch È stato rilasciato Logs Insights	È possibile utilizzare CloudWatch Logs Insights per cercare e analizzare in modo interattivo i dati di registro. Per ulteriori informazioni, consulta Analizza i dati di log con CloudWatch Logs Insights nella Amazon CloudWatch Logs User Guide.	27 novembre 2018
Supporto per gli endpoint Amazon VPC	Ora puoi stabilire una connessione privata tra il tuo VPC e CloudWatch Logs. Per ulteriori informazioni, consulta Using CloudWatch Logs with Interface VPC Endpoints nella CloudWatch Amazon Logs User Guide.	28 giugno 2018

La tabella seguente descrive le modifiche importanti alla Amazon CloudWatch Logs User's Guide.

Modifica	Descrizione	Data di rilascio
Endpoint VPC di interfaccia	In alcune regioni, puoi utilizzare un endpoint VPC di interfaccia per impedire che il traffico tra Amazon VPC e CloudWatch Logs esca dalla rete Amazon. Per ulteriori informazioni, consulta Utilizzo dei CloudWatch log con endpoint VPC di interfaccia.	7 marzo 2018
Log di query DNS di Route 53	Puoi utilizzare CloudWatch Logs per archiviar e i log relativi alle query DNS ricevute da Route 53. Per ulteriori informazioni, consulta Che cos'è Amazon CloudWatch Logs? o Registrazione delle query DNS nella Guida per gli sviluppatori di Amazon Route 53.	7 settembre 2017

Modifica	Descrizione	Data di rilascio
Assegnazione di tag ai gruppi di log	Puoi utilizzare i tag per categorizzare i gruppi di log. Per ulteriori informazioni, consulta Contrassegna i gruppi di log in Amazon CloudWatch Logs.	13 dicembre 2016
Miglioramenti della console	Puoi spostarti dai grafici di parametri ai gruppo di log associati. Per ulteriori informazioni, consulta Cambiare da parametri a log.	7 Novembre 2016
Miglioramenti della fruibilità della console	Migliorata per rendere più facili la ricerche, l'applicazione di filtri e la risoluzioni dei problemi. Ad esempio, è ora possibile filtrare i dati di log per un intervallo di data e orari. Per ulteriori informazioni, consulta Visualizza i dati di registro inviati ai registri CloudWatch .	29 agosto 2016
Aggiunto AWS CloudTrail supporto per Amazon CloudWatch Logs e nuove metriche CloudWatch Logs	È stato aggiunto il AWS CloudTrail supporto per Logs. CloudWatch Per ulteriori informazioni, consultare Registrazione delle chiamate API del File di log Amazon CloudWatch in AWS CloudTrail.	10 marzo 2016
È stato aggiunto il supporto per l'esportazione CloudWatch dei log in Amazon S3	È stato aggiunto il supporto per l'esportazione dei dati dei CloudWatch log in Amazon S3. Per ulteriori informazioni, consulta Esportazione di dati di log in Amazon S3.	7 dicembre 2015

Modifica	Descrizione	Data di rilascio
Aggiunto il supporto per gli eventi AWS CloudTrai l registrati in Amazon CloudWatch Logs	Puoi creare allarmi CloudWatch e ricevere notifiche relative a particolari attività API acquisite da CloudTrail e utilizzare la notifica per eseguire la risoluzione dei problemi.	10 novembre 2014
Aggiunto il supporto per Amazon CloudWatch Logs	Puoi utilizzare Amazon CloudWatch Logs per monitorare, archiviare e accedere a sistemi, applicazioni e file di log personalizzati da istanze Amazon Elastic Compute Cloud (Amazon EC2) o altre fonti. Puoi quindi recuperare i dati di log associati da CloudWatch Logs utilizzando la CloudWatch console Amazon, i comandi CloudWatch Logs in o Logs AWS CLISDK. CloudWatch Per ulteriori informazioni, consulta Che cos'è Amazon CloudWatch Logs?.	10 luglio 2014

AWS Glossario

Per la AWS terminologia più recente, consultate il AWS glossario nella sezione Reference.Glossario AWS

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