



Guida di riferimento

# AWS SDK e strumenti



# AWS SDK e strumenti: Guida di riferimento

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

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

---

# Table of Contents

AWSGuida di riferimento agli SDK e agli strumenti .....	1
Risorse per sviluppatori .....	2
Configurazione .....	4
Condivisi config e credentials file .....	5
Profili .....	5
Formato del file di configurazione .....	6
Formato del file delle credenziali .....	9
Posizione dei file condivisi .....	10
Cambia la posizione predefinita di questi file .....	11
Variabili di ambiente .....	12
Come impostare le variabili di ambiente .....	12
Configurazione delle variabili di ambiente senza server .....	13
Proprietà del sistema JVM .....	14
Come impostare le proprietà del sistema JVM .....	14
Autenticazione e accesso .....	17
ID Builder AWS .....	18
Autenticazione IAM Identity Center .....	19
Configura l'accesso programmatico utilizzando IAM Identity Center .....	19
Comprendi l'autenticazione IAM Identity Center .....	23
IAM Roles Anywhere .....	27
Fase 1: configurare IAM Roles Anywhere .....	27
Passaggio 2: utilizza IAM Roles Anywhere .....	27
Assunzione di un ruolo .....	29
Assumi un ruolo IAM .....	29
Federazione con identità web o OpenID Connect .....	30
AWS chiavi di accesso .....	32
Usa credenziali a breve termine .....	32
Usa credenziali a lungo termine .....	32
Credenziali a breve termine .....	34
Credenziali a lungo termine .....	35
Ruoli IAM per istanze Amazon EC2 .....	38
Creazione di un ruolo IAM .....	38
Avvia un'istanza Amazon EC2 e specifica il tuo ruolo IAM .....	39
Connect all'istanza EC2 .....	39

---

Esegui l'applicazione di esempio sull'istanza EC2 .....	40
Riferimento alle impostazioni .....	41
Creazione di client di servizio .....	41
Precedenza delle impostazioni .....	41
Configelenco delle impostazioni dei file .....	42
Credentialselelenco delle impostazioni dei file .....	46
elenco delle variabili di ambiente .....	46
Elenco delle proprietà del sistema JVM .....	50
Fornitori di credenziali standardizzati .....	53
Catena di fornitori di credenziali .....	54
AWS chiavi di accesso .....	55
Assumi il ruolo di fornitore .....	58
Fornitore di contenitori .....	64
Fornitore di IAM Identity Center .....	68
Fornitore IMDS .....	74
Fornitore di processi .....	78
Funzionalità standardizzate .....	82
Metadati delle istanze Amazon EC2 .....	83
Punti di accesso Amazon S3 .....	85
Punti di accesso multi-Regione di Amazon S3 .....	87
Regione AWS .....	89
AWS STS Endpoint regionalizzati .....	92
Endpoint dual-stack e FIPS .....	95
Rilevamento di endpoint .....	97
Configurazione generale .....	99
Cliente IMDS .....	103
Comportamento di ripetizione .....	106
Richiedi la compressione .....	110
Endpoint specifici del servizio .....	112
Impostazioni predefinite di configurazione intelligenti .....	161
Common Runtime .....	166
Dipendenze CRT .....	167
Manutenzione e supporto .....	168
Politica di manutenzione .....	168
Panoramica .....	168
Controllo delle versioni .....	168

---

Ciclo di vita della versione principale dell'SDK .....	169
Ciclo di vita delle dipendenze .....	170
Metodi di comunicazione .....	170
Matrice di supporto delle versioni .....	171
Toolkit IDE .....	173
Notifica di telemetria .....	174
Cronologia dei documenti .....	175
Glossario AWS .....	178
.....	clxxix

# AWSGuida di riferimento agli SDK e agli strumenti

Molti SDK e strumenti condividono alcune funzionalità comuni, tramite specifiche di progettazione condivise o tramite una libreria condivisa.

Questa guida include informazioni riguardanti:

- [Configurazione](#)— Come utilizzare le variabili condivise `config` e di `credentials` file o di ambiente per configurare gli AWS SDK e gli strumenti.
- [Autenticazione e accesso](#)— Stabilisci in che modo il codice o lo strumento si autentica AWS durante lo sviluppo con. Servizi AWS
- [Riferimento alle impostazioni](#)— Riferimento per tutte le impostazioni standardizzate disponibili per l'autenticazione e la configurazione.
- [AWSLibrerie Common Runtime \(CRT\)](#)— Panoramica delle librerie AWS Common Runtime (CRT) condivise disponibili per quasi tutti gli SDK.
- [AWSPolitica di manutenzione degli SDK e degli strumenti](#) copre la politica di manutenzione e il controllo delle versioni per i AWS Software Development Kit (SDK) e gli strumenti, inclusi gli SDK per dispositivi mobili e Internet of Things (IoT), e le relative dipendenze sottostanti.

Questa guida di riferimento agli AWS SDK e agli strumenti vuole essere una base di informazioni applicabile a più SDK e strumenti. La guida specifica per l'SDK o lo strumento che stai utilizzando deve essere utilizzata in aggiunta a tutte le informazioni qui presentate. Di seguito sono riportati l'SDK e gli strumenti che contengono sezioni di materiale pertinenti in questa guida:

Se stai usando:	Le sezioni pertinenti di questa guida per te sono:
<ul style="list-style-type: none"> <li>• Qualsiasi SDK o strumento</li> </ul>	<a href="#">AWSPolitica di manutenzione degli SDK e degli strumenti</a>
<ul style="list-style-type: none"> <li>• <a href="#">Guida per sviluppatori di AWS Cloud Development Kit (AWS CDK)</a></li> <li>• <a href="#">Guida per sviluppatori di AWS Serverless Application Model</a></li> <li>• <a href="#">Guida per l'utente di AWS Toolkit for Eclipse</a></li> </ul>	<a href="#">Configurazione</a> <a href="#">Autenticazione e accesso</a> <a href="#">AWSPolitica di manutenzione degli SDK e degli strumenti</a>

Se stai usando:	Le sezioni pertinenti di questa guida per te sono:
<ul style="list-style-type: none"><li>• <a href="#">Guida per l'utente di AWS Toolkit for JetBrains</a></li><li>• <a href="#">Guida per l'utente di AWS Toolkit for Visual Studio</a></li><li>• <a href="#">Guida per l'utente di AWS Toolkit for Visual Studio Code</a></li></ul>	
<ul style="list-style-type: none"><li>• <a href="#">Guida per l'utente di AWS Command Line Interface</a></li><li>• <a href="#">Guida per sviluppatori di AWS SDK for C++</a></li><li>• <a href="#">Guida per sviluppatori di AWS SDK for Go</a></li><li>• <a href="#">Guida per sviluppatori di AWS SDK for Java</a></li><li>• <a href="#">Guida per sviluppatori di AWS SDK for JavaScript</a></li><li>• <a href="#">SDK AWS for Kotlin</a></li><li>• <a href="#">Guida per sviluppatori di AWS SDK for .NET</a></li><li>• <a href="#">Guida per sviluppatori di AWS SDK for PHP</a></li><li>• <a href="#">AWSSDK per Python (Boto3) Guida introduttiva</a></li><li>• <a href="#">Guida per sviluppatori di AWS SDK for Ruby</a></li><li>• <a href="#">AWS SDK for Rust</a></li><li>• <a href="#">SDK AWS per Swift</a></li><li>• <a href="#">Guida per l'utente di AWS Tools for Windows PowerShell</a></li></ul>	<ul style="list-style-type: none"><li>• <a href="#">Configurazione</a></li><li>• <a href="#">Autenticazione e accesso</a></li><li>• <a href="#">Riferimento alle impostazioni</a></li><li>• <a href="#">AWSLibrerie Common Runtime (CRT)</a></li><li>• <a href="#">AWSPolitica di manutenzione degli SDK e degli strumenti</a></li></ul>

## Risorse per sviluppatori

Amazon CodeWhisperer è un servizio basato sull'apprendimento automatico (ML) che aiuta a migliorare la produttività degli sviluppatori generando consigli sul codice basati sia sui commenti del codice che sul codice nell'ambiente di sviluppo integrato (IDE). Per ulteriori informazioni su

quali lingue e IDE sono supportati e su come registrarsi per l'anteprima gratuita, consulta [Amazon CodeWhisperer](#).



# Configurazione

Con gli AWS SDK e altri strumenti di AWS sviluppo, come AWS Command Line Interface (AWS CLI), puoi interagire con le API di AWS servizio. Prima di eseguire questa operazione, tuttavia, è necessario configurare l'SDK o lo strumento con le informazioni necessarie per eseguire l'operazione richiesta.

Queste informazioni includono i seguenti elementi:

- Informazioni sulle credenziali che identificano chi sta chiamando l'API. Le credenziali vengono utilizzate per crittografare la richiesta ai server. AWS Utilizzando queste informazioni, AWS conferma la tua identità e puoi recuperare le politiche di autorizzazione ad essa associate. Quindi può determinare quali azioni sei autorizzato a eseguire.
- Altri dettagli di configurazione che usi per indicare all' AWS CLI SDK come elaborare la richiesta, dove inviare la richiesta (a quale endpoint del AWS servizio) e come interpretare o visualizzare la risposta.

Ogni SDK o strumento supporta più fonti che puoi utilizzare per fornire le credenziali e le informazioni di configurazione richieste. Alcune fonti sono esclusive dell'SDK o dello strumento e devi fare riferimento alla documentazione di tale strumento o SDK per i dettagli su come utilizzare tale metodo.

Tuttavia, la maggior parte degli AWS SDK e degli strumenti supporta impostazioni comuni provenienti da due fonti principali (oltre al codice stesso):

- File di [AWS configurazione e credenziali condivisi: i `credentials file`](#) `config` e `condivisi` sono il modo più comune per specificare l'autenticazione e la configurazione su un AWS SDK o uno strumento. Usa questi file per archiviare le impostazioni utilizzabili dai tuoi strumenti e dalle tue applicazioni. Le impostazioni all'interno `credentials` dei file `condivisi config` sono associate a un profilo specifico. Con più profili, puoi creare diverse configurazioni di impostazioni da applicare in diversi scenari. Quando utilizzate AWS uno strumento per richiamare un comando o utilizzate un SDK per richiamare un' AWS API, potete specificare quale profilo, e quindi quali impostazioni di configurazione, utilizzare per quell'azione. Uno dei profili è designato come `default` profilo e viene utilizzato automaticamente quando non si specifica esplicitamente un profilo da utilizzare. Le impostazioni che è possibile memorizzare in questi file sono documentate in questa guida di riferimento.
- [Variabili di ambiente](#): in alternativa, alcune impostazioni possono essere memorizzate nelle variabili di ambiente del sistema operativo. Sebbene sia possibile avere un solo set di variabili di ambiente

attive alla volta, queste possono essere facilmente modificate dinamicamente man mano che il programma viene eseguito e i requisiti cambiano.

Argomenti aggiuntivi in questa sezione

- [Condivisi config e credentials file](#)
- [Ubicazione degli elementi condivisi config e credentials dei file](#)
- [Supporto per variabili di ambiente](#)
- [Supporto delle proprietà del sistema JVM](#)

## Condivisi **config** e **credentials** file

I `credentials` file condivisi AWS `config` e contengono un set di profili. Un profilo è un insieme di valori di configurazione a cui è possibile fare riferimento dall'SDK/Tool utilizzando il nome del profilo. I valori di configurazione sono allegati a un profilo per configurare alcuni aspetti dell'SDK/strumento quando viene utilizzato quel profilo. Questi file sono «condivisi» in quanto i valori hanno effetto su qualsiasi applicazione, processo o SDK nell'ambiente locale di un utente.

Come regola generale, qualsiasi valore che è possibile inserire nel `credentials` file condiviso può essere inserito in alternativa nel `config` file condiviso. Non è vero il contrario; nel `credentials` file possono essere inserite solo alcune impostazioni. Tuttavia, come procedura consigliata in materia di sicurezza, consigliamo di conservare tutti i valori sensibili, come gli ID delle chiavi di accesso e le chiavi segrete, in un `credentials` file separato. In questo modo, puoi fornire autorizzazioni separate per ogni file, se necessario.

Sia i file condivisi che `config` i `credentials` file sono file di testo semplice che contengono solo caratteri ASCII (con codifica UTF-8). [Assumono la forma di quelli che vengono generalmente definiti file INI.](#)

### Profili

Le impostazioni all'interno dei `credentials` file `config` e dei file condivisi sono associate a un profilo specifico. Con più profili, puoi creare diverse configurazioni di impostazioni da applicare in diversi scenari.

Il `[default]` profilo contiene i valori utilizzati da un SDK o dal funzionamento dello strumento se non viene specificato un profilo denominato specifico. Puoi anche creare profili separati a cui

puoi fare riferimento esplicitamente per nome. Ogni profilo denominato può avere un gruppo di impostazioni diverso.

[default] è semplicemente un profilo senza nome. Questo profilo è denominato default perché è il profilo predefinito utilizzato dall'SDK se l'utente non specifica un profilo. Non fornisce valori predefiniti ereditati ad altri profili. Ad esempio, se si imposta qualcosa nel [default] profilo e non lo si imposta in un profilo denominato, il valore non viene impostato quando si utilizza il profilo denominato.

## Imposta un profilo denominato

Invece di usare il [default] profilo, puoi impostare un profilo con nome.

Configura questa funzionalità impostando una delle seguenti opzioni:

### **AWS\_PROFILE**- variabile di ambiente

Tutto il codice e AWS CLI i comandi SDK utilizzano le impostazioni di questo profilo denominato.

Esempio in Linux/macOS di impostazione delle variabili di ambiente tramite riga di comando:

```
export AWS_PROFILE="my_default_profile_name";
```

Esempio in Windows di impostazione delle variabili di ambiente tramite riga di comando:

```
setx AWS_PROFILE "my_default_profile_name"
```

### **aws.profile**- Proprietà del sistema JVM

[Per gli SDK eseguiti su una JVM, puoi impostare la proprietà del sistema. `aws.profile`](#) Quando l'SDK crea un client di servizio, utilizza le impostazioni del profilo denominato, a meno che l'impostazione non venga sovrascritta nel codice.

## Formato del file di configurazione

Il `config` file è organizzato in sezioni. Una sezione è una raccolta denominata di impostazioni e continua fino a quando non viene incontrata un'altra riga di definizione della sezione.

Il `config` file è un file di testo semplice che utilizza il seguente formato:

- Tutte le voci di una sezione assumono la forma generale di `setting-name=value`
- Le righe possono essere commentate iniziando la riga con un carattere hashtag (`#`).

## Tipi di sezione

Una definizione di sezione è una riga che applica un nome a una raccolta di impostazioni. Le linee di definizione della sezione iniziano e finiscono con parentesi quadre (`[]`). All'interno delle parentesi, c'è un identificatore del tipo di sezione e un nome personalizzato per la sezione. È possibile utilizzare lettere, numeri, trattini (`-`) e caratteri di sottolineatura (`_`), ma non spazi.

Tipo di sezione: **profile**

Esempio di riga di definizione della sezione: `[profile dev]`

La riga di definizione della `profile` sezione indica un raggruppamento di configurazione che è possibile applicare in diversi scenari. `[default]` è l'unico profilo che non richiede l'identificatore di `profile` sezione. Per comprendere meglio i profili denominati, consultate la sezione precedente sui profili.

L'esempio seguente mostra un config file di base con un `[default]` profilo. Imposta l'[region](#) impostazione.

```
[default]
#Full line comment, this text is ignored.
region = us-east-2
```

L'esempio seguente mostra un config file con una riga di definizione della `profile` sezione. Utilizza l'identificatore `profile` seguito da un nome univoco per il profilo. Tutte le impostazioni che seguono questa riga, fino alla definizione di un'altra sezione, verranno incluse in questo profilo denominato.

```
[profile developers]
...settings...
```

Alcune impostazioni hanno un proprio gruppo annidato di sottoimpostazioni, come l'`s3` impostazione e le sottoimpostazioni dell'esempio seguente. Associate le sottoimpostazioni al gruppo facendole rientrare con uno o più spazi.

```
[profile testers]
```

```
region = us-west-2
s3 =
    max_concurrent_requests=10
    max_queue_size=1000
```

Tipo di sezione: **sso-session**

Esempio di riga di definizione della sezione: `[sso-session my-sso]`

La riga di definizione della `sso-session` sezione nomina un gruppo di impostazioni utilizzate per configurare un profilo con cui risolvere AWS le credenziali. AWS IAM Identity Center Per ulteriori informazioni sulla configurazione dell'autenticazione Single Sign-On, vedere. [Autenticazione IAM Identity Center](#) Un profilo è collegato a una `sso-session` sezione da una coppia chiave-valore in cui `sso-session` è la chiave e il nome della `sso-session` sezione è il valore, ad esempio. `sso-session = <name-of-sso-session-section>`

L'esempio seguente configura un profilo che otterrà AWS le credenziali a breve termine per il ruolo "SampleRole" IAM nell'account «111122223333» utilizzando un token proveniente da «my-sso». La sezione «my-sso» viene referenziata `sso-session` nella sezione per nome utilizzando la chiave. `profile sso-session`

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

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

Tipo di sezione: **services**

Esempio di riga di definizione della sezione: `[services dev]`

### Note

La `services` sezione supporta le personalizzazioni degli endpoint specifici del servizio ed è disponibile solo negli SDK e negli strumenti che includono questa funzionalità. Per vedere se questa funzionalità è disponibile per il tuo SDK, consulta la sezione dedicata agli endpoint specifici del servizio. [Compatibilità con AWS gli SDK](#)

La riga di definizione della `services` sezione indica un gruppo di impostazioni che configurano endpoint personalizzati per le richieste. Servizio AWS Un profilo è collegato a una `services` sezione tramite una coppia chiave-valore in cui `services` è la chiave e il nome della `services` sezione è il valore, ad esempio. `services = <name-of-services-section>`

La `services` sezione è ulteriormente suddivisa in sottosezioni mediante `<SERVICE> =` righe, dove si `<SERVICE>` trova la Servizio AWS chiave identificativa. L' Servizio AWS identificatore si basa sul modello API e sostituisce tutti gli spazi con caratteri `serviceId` di sottolineatura e tutte le lettere minuscole. Per un elenco di tutte le chiavi identificative del servizio da utilizzare nella sezione, vedere. `services` [Identificatori per endpoint specifici del servizio](#) La chiave identificativa del servizio è seguita da impostazioni annidate, ciascuna sulla propria riga e rientrata da due spazi.

L'esempio seguente utilizza una `services` definizione per configurare l'endpoint da utilizzare per le richieste effettuate solo al servizio. Amazon DynamoDB La `"local-dynamodb"` `services` sezione viene referenziata nella `profile` sezione per nome utilizzando la `services` chiave. La chiave Servizio AWS identificativa è. `dynamodb` La sottosezione del Amazon DynamoDB servizio inizia sulla linea. `dynamodb =` Tutte le righe immediatamente successive che sono rientrate sono incluse in tale sottosezione e si applicano a quel servizio.

```
[profile dev]
services = local-dynamodb

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

Per ulteriori informazioni sulla configurazione personalizzata degli endpoint, vedere. [Endpoint specifici del servizio](#)

## Formato del file delle credenziali

Le regole per il `credentials` file sono generalmente identiche a quelle del `config` file, tranne per il fatto che le sezioni del profilo non iniziano con la `parolaprofile`. Usa solo il nome univoco del profilo stesso tra parentesi quadre.

```
[dev]
...settings...
```



2. (Piattaforme Windows) La variabile di USERPROFILE ambiente
3. (Piattaforme Windows) La variabile di HOMEDRIVE ambiente, aggiunta alla variabile di HOMEPATH ambiente (ad esempio,) \$HOMEDRIVE\$HOMEPATH
4. (Facoltativa per SDK o strumento) Una funzione o variabile di risoluzione del percorso home specifica dell'SDK o dello strumento

Quando possibile, se la home directory di un utente viene specificata all'inizio del percorso (ad esempio,~username/), viene risolta nella home directory del nome utente richiesto (ad esempio,) /home/username/.aws/config

## Cambia la posizione predefinita di questi file

### Usa variabili di ambiente

Le seguenti variabili di ambiente possono essere impostate per modificare la posizione o il nome di questi file dal valore predefinito a un valore personalizzato:

- configvariabile di ambiente del file: **AWS\_CONFIG\_FILE**
- credentialsvariabile di ambiente di file: **AWS\_SHARED\_CREDENTIALS\_FILE**

### Linux/macOS

È possibile specificare una posizione alternativa eseguendo i seguenti comandi di [esportazione](#) su Linux o macOS.

```
$ export AWS_CONFIG_FILE=/some/file/path/on/the/system/config-file-name
$ export AWS_SHARED_CREDENTIALS_FILE=/some/other/file/path/on/the/system/
credentials-file-name
```

### Windows

È possibile specificare una posizione alternativa eseguendo i seguenti comandi [setx](#) su Windows.

```
C:\> setx AWS_CONFIG_FILE c:\some\file\path\on\the\system\config-file-name
C:\> setx AWS_SHARED_CREDENTIALS_FILE c:\some\other\file\path\on\the\system
\credentials-file-name
```



## Usa le proprietà del sistema JVM

È possibile impostare le seguenti proprietà del sistema JVM per modificare la posizione o il nome di questi file dal valore predefinito a un valore personalizzato:

- configproprietà del sistema JVM del file: **aws.configFile**
- credentialsvariabile di ambiente del file: **aws.sharedCredentialsFile**

Per istruzioni su come impostare le proprietà del sistema JVM, vedere. [the section called “Come impostare le proprietà del sistema JVM”](#)

## Supporto per variabili di ambiente

Le variabili di ambiente offrono un altro modo per specificare le opzioni di configurazione e le credenziali e possono essere utili per la creazione di script o l'impostazione temporanea di un profilo denominato come profilo di default. Per l'elenco delle variabili di ambiente supportate dalla maggior parte degli SDK, consulta. [elenco delle variabili di ambiente](#)

### Precedenza delle opzioni

- Se specifichi un'impostazione utilizzando la relativa variabile di ambiente, questa sovrascrive qualsiasi valore caricato da un profilo nei file condivisi AWS config e credentials
- Se specificate un'impostazione utilizzando un parametro sulla riga di AWS CLI comando, questo sovrascrive qualsiasi valore della variabile di ambiente corrispondente o di un profilo nel file di configurazione.

## Come impostare le variabili di ambiente

L'esempio seguente mostra come configurare le variabili di ambiente per l'utente predefinito.

Linux, macOS, or Unix

```
$ export AWS_ACCESS_KEY_ID=AKIAIOSFODNN7EXAMPLE
$ export AWS_SECRET_ACCESS_KEY=wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
$ export
AWS_SESSION_TOKEN=AQoEXAMPLEH4aoAH0gNCAPy...truncated...zrkuWJ0gQs8IZZaIv2BXIa2R40Lgk
$ export AWS_REGION=us-west-2
```

L'impostazione della variabile di ambiente modifica il valore utilizzato fino al termine della sessione della shell o finché non imposti la variabile su un valore diverso. Puoi rendere le variabili persistenti per le sessioni future impostandole nello script di avvio della shell.

## Windows Command Prompt

```
C:\> setx AWS_ACCESS_KEY_ID AKIAIOSFODNN7EXAMPLE
C:\> setx AWS_SECRET_ACCESS_KEY wJa1rXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
C:\> setx
AWS_SESSION_TOKEN AQoEXAMPLEH4aoAH0gNCApy...truncated...zrkuWJ0gQs8IZZaIv2BXIa2R40lgk
C:\> setx AWS_REGION us-west-2
```

L'utilizzo [set](#) per impostare una variabile di ambiente modifica il valore utilizzato fino alla fine della sessione corrente del prompt dei comandi o fino a quando non si imposta la variabile su un valore diverso. Se si utilizza [setx](#) per impostare una variabile di ambiente, viene modificato il valore utilizzato sia nella sessione corrente del prompt dei comandi che in tutte le sessioni del prompt dei comandi create dopo l'esecuzione del comando. Ciò non ha alcun impatto su altre shell di comando già in esecuzione quando esegui il comando.

## PowerShell

```
PS C:\> $Env:AWS_ACCESS_KEY_ID="AKIAIOSFODNN7EXAMPLE"
PS C:\> $Env:AWS_SECRET_ACCESS_KEY="wJa1rXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY"
PS C:
\> $Env:AWS_SESSION_TOKEN="AQoEXAMPLEH4aoAH0gNCApy...truncated...zrkuWJ0gQs8IZZaIv2BXIa2R40lgk"
PS C:\> $Env:AWS_REGION="us-west-2"
```

Se impostate una variabile di ambiente al PowerShell prompt, come mostrato negli esempi precedenti, il valore viene salvato solo per la durata della sessione corrente. Per rendere persistente l'impostazione della variabile di ambiente in tutte PowerShell le sessioni del prompt dei comandi, memorizzatela utilizzando l'applicazione System nel Pannello di controllo. In alternativa, puoi impostare la variabile per tutte le PowerShell sessioni future aggiungendola al tuo PowerShell profilo. Consulta la [PowerShell documentazione](#) per ulteriori informazioni sulla memorizzazione delle variabili di ambiente o sulla loro persistenza tra le sessioni.

## Configurazione delle variabili di ambiente senza server

Se si utilizza un'architettura serverless per lo sviluppo, sono disponibili altre opzioni per l'impostazione delle variabili di ambiente. A seconda del contenitore, puoi utilizzare diverse strategie

per l'esecuzione del codice in tali contenitori per visualizzare e accedere alle variabili di ambiente, in modo simile agli ambienti non cloud.

Ad esempio, con AWS Lambda, puoi impostare direttamente le variabili di ambiente. Per i dettagli, consulta [Uso delle variabili di AWS Lambda ambiente](#) nella Guida per AWS Lambda gli sviluppatori.

In Serverless Framework, puoi spesso impostare le variabili di ambiente SDK nel `serverless.yml` file sotto la chiave del provider sotto l'impostazione dell'ambiente. Per informazioni sul `serverless.yml` file, consulta [Impostazioni generali delle funzioni](#) nella documentazione di Serverless Framework.

Indipendentemente dal meccanismo utilizzato per impostare le variabili di ambiente del contenitore, ce ne sono alcune riservate dal contenitore, come quelle documentate per Lambda nelle variabili di ambiente di [runtime definite](#). Consulta sempre la documentazione ufficiale del contenitore che stai utilizzando per determinare come vengono trattate le variabili di ambiente e se esistono restrizioni.

## Supporto delle proprietà del sistema JVM

[Le proprietà del sistema JVM](#) forniscono un altro modo per specificare le opzioni di configurazione e le credenziali per gli SDK che vengono eseguiti sulla JVM, ad esempio il e il. AWS SDK for Java SDK AWS for Kotlin [Per un elenco delle proprietà del sistema JVM supportate dagli SDK, consulta il riferimento alle impostazioni.](#)

### Precedenza delle opzioni

- Se specifichi un'impostazione utilizzando la relativa proprietà di sistema JVM, questa sovrascrive qualsiasi valore trovato nelle variabili di ambiente o caricato da un profilo in AWS e file condivisi. `config credentials`
- Se specifichi un'impostazione utilizzando la relativa variabile di ambiente, questa sovrascrive qualsiasi valore caricato da un profilo nei file `config` e `credentials` nei file AWS condivisi.

## Come impostare le proprietà del sistema JVM

È possibile impostare le proprietà del sistema JVM in diversi modi.

## Sulla riga di comando

Imposta le proprietà del sistema JVM sulla riga di comando quando richiami il `java` comando utilizzando lo switch. `-D` Il comando seguente configura Regione AWS globalmente per tutti i client di servizio a meno che non si sovrascriva esplicitamente il valore nel codice.

```
java -Daws.region=us-east-1 -jar <your_application.jar> <other_arguments>
```

Se è necessario impostare più proprietà del sistema JVM, specificare lo switch più volte. `-D`

## Con una variabile di ambiente

Se non riesci ad accedere alla riga di comando per richiamare la JVM per eseguire l'applicazione, puoi utilizzare la variabile di `JAVA_TOOL_OPTIONS` ambiente per configurare le opzioni della riga di comando. Questo approccio è utile in situazioni come l'esecuzione di una AWS Lambda funzione sul runtime Java o l'esecuzione di codice in una JVM incorporata.

L'esempio seguente configura Regione AWS globalmente per tutti i client di servizio a meno che non si sovrascriva esplicitamente il valore nel codice.

Linux, macOS, or Unix

```
$ export JAVA_TOOL_OPTIONS="-Daws.region=us-east-1"
```

L'impostazione della variabile di ambiente modifica il valore utilizzato fino al termine della sessione della shell o finché non imposti la variabile su un valore diverso. Puoi rendere le variabili persistenti per le sessioni future impostandole nello script di avvio della shell.

Windows Command Prompt

```
C:\> setx JAVA_TOOL_OPTIONS -Daws.region=us-east-1
```

L'utilizzo [set](#) per impostare una variabile di ambiente modifica il valore utilizzato fino alla fine della sessione corrente del prompt dei comandi o fino a quando non si imposta la variabile su un valore diverso. Se si utilizza [setx](#) per impostare una variabile di ambiente, viene modificato il valore utilizzato sia nella sessione corrente del prompt dei comandi che in tutte le sessioni del prompt dei comandi create dopo l'esecuzione del comando. Ciò non ha alcun impatto su altre shell di comando già in esecuzione quando esegui il comando.

## In fase di esecuzione

È inoltre possibile impostare le proprietà del sistema JVM in fase di esecuzione nel codice utilizzando il `System.setProperty` metodo illustrato nell'esempio seguente.

```
System.setProperty("aws.region", "us-east-1");
```

### Important

Impostate le proprietà del sistema JVM prima di inizializzare i client del servizio SDK, altrimenti i client di servizio potrebbero utilizzare altri valori.

# Autenticazione e accesso

È necessario stabilire in che modo il codice si autentica AWS durante lo sviluppo con. Servizi AWS È possibile configurare l'accesso programmatico alle AWS risorse in diversi modi, a seconda dell'ambiente e dell'AWSaccesso a disposizione.

Opzioni di autenticazione per il codice eseguito localmente (non inAWS)

- [Autenticazione IAM Identity Center](#)— Come best practice di sicurezza, ti consigliamo di utilizzare AWS Organizations IAM Identity Center per gestire l'accesso su tutti i tuoi dispositiviAccount AWS. Puoi creare utenti inAWS IAM Identity Center, utilizzare Microsoft Active Directory, utilizzare un provider di identità (IdP) SAML 2.0 o federare individualmente il tuo IdP in. Account AWS Per verificare se la tua regione supporta IAM Identity Center, consulta gli [AWS IAM Identity Centerendpoint](#) e le quote nel. Riferimenti generali di Amazon Web Services
- [IAM Roles Anywhere](#)— Puoi utilizzare IAM Roles Anywhere per ottenere credenziali di sicurezza temporanee in IAM per carichi di lavoro come server, contenitori e applicazioni eseguiti all'esterno di. AWS Per utilizzare IAM Roles Anywhere, i carichi di lavoro devono utilizzare certificati X.509.
- [Assunzione di un ruolo](#)— Puoi assumere un ruolo IAM per accedere temporaneamente a AWS risorse a cui altrimenti non avresti accesso.
- [AWS chiavi di accesso](#)— Altre opzioni che potrebbero essere meno convenienti o che potrebbero aumentare il rischio di sicurezza per le AWS risorse.

Opzioni di autenticazione per il codice in esecuzione all'interno di un AWS ambiente

- [Utilizzo dei ruoli IAM per le istanze Amazon EC2](#)— Usa i ruoli IAM per eseguire in modo sicuro la tua applicazione su un'istanza Amazon EC2.
- Puoi interagire a livello di codice con l'AWSutilizzo di IAM Identity Center nei seguenti modi:
  - [AWS CloudShell](#)Da utilizzare per eseguire AWS CLI comandi dalla console.
  - [AWS Cloud9](#)Da utilizzare per iniziare a programmare AWS utilizzando un ambiente di sviluppo integrato (IDE) con AWS risorse.
  - [Per provare uno spazio di collaborazione basato sul cloud per i team di sviluppo software, prendi in considerazione l'utilizzo di Amazon. CodeCatalyst](#)

Autenticazione tramite un provider di identità basato sul Web: applicazioni Web mobili o basate su client

Se stai creando applicazioni mobili o applicazioni web basate su client che richiedono l'accesso aAWS, crea la tua app in modo che richieda le credenziali di AWS sicurezza temporanee in modo dinamico utilizzando la federazione delle identità web.

Con la federazione delle identità Web, non è necessario creare il codice di accesso personalizzato o gestire le identità utente personalizzate. Gli utenti dell'app possono invece accedere utilizzando un provider di identità (IdP) esterno noto, come Login with Amazon, Facebook, Google o qualsiasi altro IdP compatibile con OpenID Connect (OIDC). Possono ricevere un token di autenticazione e scambiarsi quindi tale token per le credenziali di sicurezza temporanee in AWS che mappano a un ruolo IAM con autorizzazioni per utilizzare le risorse nel tuo Account AWS.

Per informazioni su come configurarlo per il tuo SDK o il tuo strumento, consulta [Federazione con identità web o OpenID Connect](#)

Per le applicazioni mobili, prendi in considerazione l'utilizzo di Amazon Cognito. Amazon Cognito funge da broker di identità e svolge gran parte del lavoro federativo per te. Per ulteriori informazioni, consulta [Using Amazon Cognito per app mobili](#) nella IAM User Guide.

Ulteriori informazioni sulla gestione degli accessi

La Guida per l'utente IAM contiene le seguenti informazioni sul controllo sicuro dell'accesso alle AWS risorse:

- [Identità IAM \(utenti, gruppi di utenti e ruoli\)](#): scopri le basi delle identità in. AWS
- [Best practice di sicurezza in IAM: raccomandazioni di sicurezza da seguire quando si sviluppano AWS applicazioni secondo il modello di responsabilità condivisa.](#)

Riferimenti generali di Amazon Web ServicesHa le basi fondamentali su quanto segue:

- [Comprensione e acquisizione AWS delle credenziali](#): accedi alle opzioni chiave e alle pratiche di gestione sia per l'accesso da console che per quello programmatico.

## ID Builder AWS

I ID Builder AWS complementi Account AWS che possiedi già o che desideri creare. Sebbene un Account AWS funga da contenitore per AWS le risorse che crei e fornisca un limite di sicurezza per tali risorse, il tuo ID Builder AWS rappresenta come individuo. Puoi accedere con il tuo ID Builder AWS per accedere a strumenti e servizi per sviluppatori come Amazon CodeWhisperer e Amazon CodeCatalyst.

- [Accedi tramite la ID Builder AWS](#) Guida per l'Accedi ad AWS utente: scopri come creare e utilizzare un ID Builder ID Builder AWS e scopri cosa offre.
- [Autenticazione con CodeWhisperer e Kit di strumenti AWS - Builder ID](#) nella Guida per l'CodeWhisperer utente: scopri come utilizza un. CodeWhisperer ID Builder AWS
- [CodeCatalystconcetti - ID Builder AWS](#) nella Amazon CodeCatalyst User Guide - Scopri come CodeCatalyst usa unID Builder AWS.

## Autenticazione IAM Identity Center

AWS IAM Identity Center è il metodo consigliato per fornire AWS credenziali quando si sviluppa su un servizio non diAWS elaborazione. Ad esempio, questo potrebbe essere qualcosa di simile al vostro ambiente di sviluppo locale. Se stai sviluppando su una AWS risorsa, come Amazon Elastic Compute Cloud (Amazon EC2) AWS Cloud9oppure, ti consigliamo di ottenere le credenziali da quel servizio.

In questo tutorial, stabilisci l'accesso a IAM Identity Center e lo configurerai per il tuo SDK o strumento utilizzando il portale di AWS accesso e il. AWS CLI

- Il portale di AWS accesso è il luogo web in cui accedi manualmente a IAM Identity Center. Il formato dell'URL è `d-xxxxxxxxxx.awsapps.com/start` *oyour\_subdomain*.awsapps.com/start. Una volta effettuato l' AWS accesso al portale di accesso, è possibile visualizzare Account AWS i ruoli configurati per quell'utente. Questa procedura utilizza il portale di AWS accesso per ottenere i valori di configurazione necessari per il processo di autenticazione SDK/Tool.
- AWS CLI Viene utilizzato per configurare l'SDK o lo strumento per utilizzare l'autenticazione IAM Identity Center per le chiamate API effettuate dal codice. Questo processo monouso aggiorna il AWS config file condiviso, che viene poi utilizzato dal tuo SDK o dallo strumento quando esegui il codice.

## Configura l'accesso programmatico utilizzando IAM Identity Center

### Fase 1: Stabilire l'accesso e selezionare il set di autorizzazioni appropriato

Se non hai ancora abilitato IAM Identity Center, consulta [Enabling IAM Identity Center](#) nella Guida per l'AWS IAM Identity Center utente.

Scegli uno dei seguenti metodi per accedere alle tue AWS credenziali.



## Non ho stabilito l'accesso tramite IAM Identity Center

1. Aggiungi un utente e aggiungi le autorizzazioni amministrative seguendo la procedura [Configure user access with the default IAM Identity Center directory](#) nella Guida per l'AWS IAM Identity Center utente.
2. Il set di `AdministratorAccess` autorizzazioni non deve essere utilizzato per lo sviluppo regolare. Si consiglia invece di utilizzare il set di `PowerUserAccess` autorizzazioni predefinito, a meno che il datore di lavoro non abbia creato un set di autorizzazioni personalizzato per questo scopo.

Segui nuovamente la stessa procedura [Configure user access with the IAM Identity Center directory predefinita](#), ma questa volta:

- Invece di creare il *Admin team* gruppo, crea un *Dev team* gruppo e sostituiscilo successivamente nelle istruzioni.
- È possibile utilizzare l'utente esistente, ma l'utente deve essere aggiunto al nuovo *Dev team* gruppo.
- Invece di creare il set di *AdministratorAccess* autorizzazioni, create un set di *PowerUserAccess* autorizzazioni e sostituitelo successivamente nelle istruzioni.

Al termine, è necessario disporre di quanto segue:

- Un `Dev team` gruppo.
  - Un set di `PowerUserAccess` autorizzazioni allegato al `Dev team` gruppo.
  - L'utente è stato aggiunto al `Dev team` gruppo.
3. Esci dal portale e accedi nuovamente per visualizzare le tue opzioni Account AWS e per `Administrator` o `PowerUserAccess`. Seleziona `PowerUserAccess` quando lavori con il tuo strumento/SDK.

Ho già accesso AWS tramite un provider di identità federato gestito dal mio datore di lavoro (come Microsoft Entra o Okta)

Accedi AWS tramite il portale del tuo provider di identità. Se il tuo amministratore cloud ti ha concesso le autorizzazioni `PowerUserAccess` (sviluppatore), vedi quelle a Account AWS cui hai accesso e il tuo set di autorizzazioni. Accanto al nome del set di autorizzazioni, vengono visualizzate

le opzioni per accedere agli account manualmente o programmaticamente utilizzando quel set di autorizzazioni.

Le implementazioni personalizzate potrebbero dare luogo a esperienze diverse, ad esempio nomi di set di autorizzazioni diversi. Se non sei sicuro del set di autorizzazioni da utilizzare, contatta il tuo team IT per ricevere assistenza.

Ho già accesso AWS tramite il portale di AWS accesso gestito dal mio datore di lavoro

Accedi AWS tramite il portale di AWS accesso. Se il tuo amministratore cloud ti ha concesso le autorizzazioni `PowerUserAccess` (sviluppatore), vedi quelle a Account AWS cui hai accesso e il tuo set di autorizzazioni. Accanto al nome del set di autorizzazioni, vengono visualizzate le opzioni per accedere agli account manualmente o programmaticamente utilizzando quel set di autorizzazioni.

Ho già accesso AWS tramite un provider di identità personalizzato federato gestito dal mio datore di lavoro

Contatta il tuo team IT per ricevere assistenza.

## Passaggio 2: configura gli SDK e gli strumenti per utilizzare IAM Identity Center

1. Sul tuo computer di sviluppo, installa la versione più recente AWS CLI.
  - a. Vedi [Installazione o aggiornamento della versione più recente di AWS CLI](#) nella Guida AWS Command Line Interface per l'utente.
  - b. (Facoltativo) Per verificare che funzioni, apri il prompt dei comandi ed esegui il `aws --version` comando. AWS CLI
2. Accedere al portale di AWS accesso. Il tuo datore di lavoro può fornire questo URL o riceverlo tramite e-mail dopo la Fase 1: Stabilisci l'accesso. In caso contrario, puoi trovare l'URL del tuo portale di AWS accesso nella dashboard di <https://console.aws.amazon.com/singlesignon/>.
  - a. Nel portale di AWS accesso, seleziona il set di autorizzazioni appropriato, quindi seleziona la riga di comando o il collegamento di accesso programmatico. Utilizza il set di `PowerUserAccess` autorizzazioni predefinito o qualsiasi set di autorizzazioni creato da te o dal tuo datore di lavoro per applicare le autorizzazioni con privilegi minimi per lo sviluppo.
  - b. Nella finestra di dialogo Ottieni credenziali, scegli macOS e Linux o Windows, a seconda del sistema operativo in uso.
  - c. Scegli il metodo di credenziali IAM Identity Center per ottenere i `SSO_Region` valori `SSO_StartURL` e di cui hai bisogno per il passaggio successivo.

3. Nel prompt dei AWS CLI comandi, esegui il `aws configure sso` comando. Quando richiesto, inserisci i valori di configurazione raccolti nel passaggio precedente. Per i dettagli su questo AWS CLI comando, consulta [Configurare il profilo con la `aws configure sso` procedura guidata](#).
  - Per il nome del profilo CLI, ti consigliamo di inserire il *valore predefinito* quando inizi. Per informazioni su come impostare profili non predefiniti (denominati) e la variabile di ambiente associata, consulta. [Profili](#)
4. (Facoltativo) Nel AWS CLI prompt dei comandi, confermate l'identità della sessione attiva eseguendo il `aws sts get-caller-identity` comando. La risposta dovrebbe mostrare il set di autorizzazioni IAM Identity Center che hai configurato.
5. Se utilizzi un AWS SDK, crea un'applicazione per il tuo SDK nel tuo ambiente di sviluppo.
  - a. Per alcuni SDK, è SS00IDC necessario aggiungere pacchetti aggiuntivi come SSO e all'applicazione prima di poter utilizzare l'autenticazione IAM Identity Center. Per i dettagli, consulta il tuo SDK specifico.
  - b. Se in precedenza hai configurato l'accesso a AWS, esamina il `AWS credentials` file condiviso per verificarne l'eventuale [AWS chiavi di accesso](#) presenza. È necessario rimuovere tutte le credenziali statiche prima che l'SDK o lo strumento utilizzino le credenziali IAM Identity Center a causa della precedenza. [Catena di fornitori di credenziali](#)

Per un'analisi approfondita del modo in cui gli SDK e gli strumenti utilizzano e aggiornano le credenziali utilizzando questa configurazione, consulta. [Comprendi l'autenticazione IAM Identity Center](#)

A seconda della durata delle sessioni configurate, l'accesso alla fine scadrà e l'SDK o lo strumento riscontreranno un errore di autenticazione. Per aggiornare nuovamente la sessione del portale di accesso quando necessario, usa il comando AWS CLI per eseguire il comando. `aws sso login`

È possibile estendere sia la durata della sessione del portale di accesso IAM Identity Center sia la durata della sessione del set di autorizzazioni. In questo modo si allunga il periodo di tempo in cui è possibile eseguire il codice prima di dover accedere nuovamente manualmente con. AWS CLI Per ulteriori informazioni, consulta gli argomenti seguenti nella Guida per l'utente AWS IAM Identity Center :

- Durata della sessione di IAM Identity Center: [configura la durata delle sessioni del portale di accesso degli utenti AWS](#)

- Durata della sessione del set di autorizzazioni: imposta la durata [della sessione](#)

Per i dettagli su tutte le impostazioni del provider IAM Identity Center per SDK e strumenti, [Provider di credenziali IAM Identity Center](#) consulta questa guida.

## Comprendi l'autenticazione IAM Identity Center

### Termini pertinenti di IAM Identity Center

I seguenti termini aiutano a comprendere il processo e la configurazione alla base AWS IAM Identity Center. La documentazione per le API AWS SDK utilizza nomi diversi da IAM Identity Center per alcuni di questi concetti di autenticazione. È utile conoscere entrambi i nomi.

La tabella seguente mostra come i nomi alternativi si relazionano tra loro.

Nome IAM Identity Center	Nome dell'API SDK	Description
Identity Center	sso	Sebbene AWS Single Sign-On venga rinominato, i namespace delle sso API manterranno il nome originale per motivi di compatibilità con le versioni precedenti. Per ulteriori informazioni, consulta la ridenominazione di <a href="#">IAM Identity Center</a> nella Guida per l'utente. AWS IAM Identity Center
Console IAM Identity Center Console amministrativa		La console che usi per configurare il Single Sign-On.
AWS accedere all'URL del portale		Un URL univoco per il tuo account IAM Identity Center, ad esempio <code>https://xxx.awsapps.</code>

Nome IAM Identity Center	Nome dell'API SDK	Description
		<code>com/start</code> . Accedi a questo portale utilizzando le tue credenziali di accesso IAM Identity Center.
Sessione IAM Identity Center Access Portal	Sessione di autenticazione	Fornisce un token di accesso al portatore al chiamante.
Sessione con set di autorizzazioni		La sessione IAM che l'SDK utilizza internamente per effettuare le Servizio AWS chiamate. Nelle discussioni informali, potresti vederla erroneamente definita «sessione di ruolo».
Credenziali del set di autorizzazioni	AWSCredenziali credenziali sigv4	Le credenziali che l'SDK utilizza effettivamente per la maggior parte delle Servizio AWS chiamate (in particolare, tutte le chiamate sigv4). Servizio AWS Nelle discussioni informali, potreste vederle erroneamente chiamate «credenziali di ruolo».
Provider di credenziali IAM Identity Center	Provider di credenziali SSO	Come si ottengono le credenziali, ad esempio la classe o il modulo che fornisce la funzionalità.

## Comprendi la risoluzione delle credenziali SDK per Servizi AWS

L'API IAM Identity Center scambia le credenziali del token del portatore con credenziali sigv4. La maggior parte Servizi AWS sono API sigv4, con alcune eccezioni come e. Amazon CodeWhisperer Amazon CodeCatalyst Di seguito viene descritto il processo di risoluzione delle credenziali utilizzato

per supportare la maggior parte delle Servizio AWS chiamate per il codice dell'applicazione. AWS IAM Identity Center

### Avvio di una sessione del portale di accesso AWS

- Inizia il processo accedendo alla sessione con le tue credenziali.
  - Usa il `aws sso login` comando in AWS Command Line Interface (AWS CLI). Questo avvia una nuova sessione di IAM Identity Center se non hai già una sessione attiva.
- Quando inizi una nuova sessione, ricevi un token di aggiornamento e un token di accesso da IAM Identity Center. AWS CLI inoltre aggiorna un file JSON della cache SSO con un nuovo token di accesso e un token di aggiornamento e lo rende disponibile per l'uso da parte degli SDK.
- Se hai già una sessione attiva, il AWS CLI comando riutilizza la sessione esistente e scadrà ogni volta che scade la sessione esistente. Per informazioni su come impostare la durata di una sessione di IAM Identity Center, consulta [Configurare la durata delle sessioni del portale di AWS accesso degli utenti nella Guida per l'utente](#). AWS IAM Identity Center
  - La durata massima della sessione è stata estesa a 90 giorni per ridurre la necessità di accessi frequenti.

### In che modo l'SDK ottiene le credenziali per le chiamate Servizio AWS

Gli SDK forniscono l'accesso a Servizi AWS quando si crea un'istanza di un oggetto client per servizio. Quando il profilo selezionato del AWS `config` file condiviso è configurato per la risoluzione delle credenziali di IAM Identity Center, IAM Identity Center viene utilizzato per risolvere le credenziali per l'applicazione.

- Il [processo di risoluzione delle credenziali](#) viene completato durante l'esecuzione, quando viene creato un client.

Per recuperare le credenziali per le API sigv4 utilizzando il single sign-on di IAM Identity Center, l'SDK utilizza il token di accesso IAM Identity Center per ottenere una sessione IAM. Questa sessione IAM è chiamata sessione del set di autorizzazioni e fornisce l'AWSaccesso all'SDK assumendo un ruolo IAM.

- La durata della sessione del set di autorizzazioni è impostata indipendentemente dalla durata della sessione di IAM Identity Center.
  - Per informazioni su come impostare la durata della sessione del set di autorizzazioni, consulta [Impostare la durata della sessione](#) nella Guida per l'AWS IAM Identity Centerutente.

- Tieni presente che le credenziali del set di autorizzazioni vengono anche chiamate credenziali e AWScredenziali sigv4 nella maggior parte della documentazione sulle API SDK. AWS

Le credenziali del set di autorizzazioni vengono restituite da una chiamata all'API IAM Identity [getRoleCredentials](#)Center all'SDK. L'oggetto client dell'SDK utilizza quel ruolo IAM assunto per effettuare chiamate a Servizio AWS, ad esempio chiedere ad Amazon S3 di elencare i bucket nel tuo account. L'oggetto client può continuare a funzionare utilizzando le credenziali del set di autorizzazioni fino alla scadenza della sessione del set di autorizzazioni.

### Scadenza e aggiornamento della sessione

Quando si utilizza il [Configurazione del provider di token SSO](#), il token di accesso orario ottenuto da IAM Identity Center viene aggiornato automaticamente utilizzando il token di aggiornamento.

- Se il token di accesso è scaduto quando l'SDK tenta di utilizzarlo, l'SDK utilizza il token di aggiornamento per cercare di ottenere un nuovo token di accesso. IAM Identity Center confronta il token di aggiornamento con la durata della sessione del portale di accesso IAM Identity Center. Se il token di aggiornamento non è scaduto, IAM Identity Center risponde con un altro token di accesso.
- Questo token di accesso può essere utilizzato per aggiornare la sessione del set di autorizzazioni dei client esistenti o per risolvere le credenziali per nuovi client.

Tuttavia, se la sessione del portale di accesso IAM Identity Center è scaduta, non viene concesso alcun nuovo token di accesso. Pertanto, la durata del set di autorizzazioni non può essere rinnovata. Scadrà (e l'accesso verrà perso) ogni volta che scade la durata della sessione del set di autorizzazioni memorizzato nella cache per i client esistenti.

Qualsiasi codice che crea un nuovo client fallirà l'autenticazione non appena scadrà la sessione di IAM Identity Center. Questo perché le credenziali del set di autorizzazioni non vengono memorizzate nella cache. Il codice non sarà in grado di creare un nuovo client e completare il processo di risoluzione delle credenziali finché non avrai un token di accesso valido.

Ricapitolando, quando l'SDK necessita di nuove credenziali del set di autorizzazioni, l'SDK verifica innanzitutto la presenza di eventuali credenziali valide ed esistenti e le utilizza. Questo vale sia che le credenziali siano per un nuovo client o per un client esistente con credenziali scadute. Se le credenziali non vengono trovate o non sono valide, l'SDK chiama l'API IAM Identity Center per ottenere nuove credenziali. Per chiamare l'API, è necessario il token di accesso. Se il token di accesso è scaduto, l'SDK utilizza il token di aggiornamento per cercare di ottenere un nuovo token di

accesso dal servizio IAM Identity Center. Questo token viene concesso se la sessione del portale di accesso IAM Identity Center non è scaduta.

## IAM Roles Anywhere

Puoi utilizzare IAM Roles Anywhere per ottenere credenziali di sicurezza temporanee in IAM per carichi di lavoro come server, contenitori e applicazioni eseguiti all'esterno di AWS. Per utilizzare IAM Roles Anywhere, i carichi di lavoro devono utilizzare certificati X.509. L'amministratore cloud deve fornire il certificato e la chiave privata necessari per configurare IAM Roles Anywhere come fornitore di credenziali.

### Fase 1: configurare IAM Roles Anywhere

IAM Roles Anywhere offre un modo per ottenere credenziali temporanee per un carico di lavoro o un processo eseguito all'esterno di AWS. Viene stabilito un trust anchor con l'autorità di certificazione per ottenere credenziali temporanee per il ruolo IAM associato. Il ruolo imposta le autorizzazioni che il carico di lavoro avrà quando il codice si autentica con IAM Roles Anywhere.

Per i passaggi per configurare trust anchor, IAM role e IAM Roles Anywhere, consulta [Creating a trust anchor and profile in Roles Anywhere nella Guida per l'utente di IAM AWS Identity and Access Management Roles Anywhere](#).

#### Note

Un profilo nella IAM Roles Anywhere User Guide si riferisce a un concetto unico all'interno del servizio IAM Roles Anywhere. Non è correlato ai profili all'interno del AWS config file condiviso.

### Passaggio 2: utilizza IAM Roles Anywhere

Per ottenere credenziali di sicurezza temporanee da IAM Roles Anywhere, utilizza lo strumento di supporto alle credenziali fornito da IAM Roles Anywhere. Lo strumento per le credenziali implementa il processo di firma per IAM Roles Anywhere.

Per istruzioni su come scaricare lo strumento di supporto alle credenziali, consulta [Ottenere credenziali di sicurezza temporanee da AWS Identity and Access Management Roles Anywhere nella Guida per l'utente di IAM Roles Anywhere](#).



Per utilizzare le credenziali di sicurezza temporanee di IAM Roles Anywhere con AWS SDK e AWS CLI, puoi configurare le `credential_process` impostazioni nel file condiviso. AWS config Gli SDK e AWS CLI supportano un provider di credenziali di processo che utilizza per l'autenticazione. `credential_process` Di seguito viene illustrata la struttura generale da impostare. `credential_process`

```
credential_process = [path to helper tool] [command] [--parameter1 value] [--parameter2 value] [...]
```

Il `credential-process` comando dello strumento di supporto restituisce credenziali temporanee in un formato JSON standard compatibile con l'impostazione. `credential_process` Notate che il nome del comando contiene un trattino, ma il nome dell'impostazione contiene un carattere di sottolineatura. Il comando richiede i seguenti parametri:

- `private-key`— Il percorso della chiave privata che ha firmato la richiesta.
- `certificate`— Il percorso del certificato.
- `role-arn`— L'ARN del ruolo per cui ottenere le credenziali temporanee.
- `profile-arn`— L'ARN del profilo che fornisce una mappatura per il ruolo specificato.
- `trust-anchor-arn`— L'ARN del trust anchor utilizzato per l'autenticazione.

L'amministratore del cloud deve fornire il certificato e la chiave privata. Tutti e tre i valori ARN possono essere copiati da. AWS Management Console L'esempio seguente mostra un `config` file condiviso che configura il recupero delle credenziali temporanee dallo strumento di supporto.

```
[profile dev]
credential_process = ./aws_signing_helper credential-process --certificate /
path/to/certificate --private-key /path/to/private-key --trust-anchor-
arn arn:aws:rolesanywhere:region:account:trust-anchor/TA_ID --profile-
arn arn:aws:rolesanywhere:region:account:profile/PROFILE_ID --role-
arn arn:aws:iam:account:role/ROLE_ID
```

Per i parametri opzionali e i dettagli aggiuntivi dello strumento di supporto, consulta [IAM Roles Anywhere Credential Helper on GitHub](#)

Per i dettagli sull'impostazione della configurazione SDK stessa e sul fornitore delle credenziali di processo, consulta questa guida. [Provider di credenziali di processo](#)

# Assunzione di un ruolo

Assumere un ruolo implica l'utilizzo di un set di credenziali di sicurezza temporanee per accedere a AWS risorse a cui altrimenti non avreste accesso. Le credenziali temporanee sono costituite da un ID chiave di accesso, una chiave di accesso segreta e un token di sicurezza. Per ulteriori informazioni sulle richieste API AWS Security Token Service (AWS STS), consulta [Azioni](#) nell'AWS Security Token ServiceAPI Reference.

Per configurare l'SDK o lo strumento per assumere un ruolo, devi prima creare o identificare un ruolo specifico da assumere. I ruoli IAM sono identificati in modo univoco da un ruolo Amazon Resource Name ([ARN](#)). I ruoli stabiliscono relazioni di fiducia con un'altra entità. L'entità affidabile che utilizza il ruolo potrebbe essere un'altra Servizio AWSAccount AWS, un provider di identità Web o una federazione OIDC o SAML. Per ulteriori informazioni sui ruoli IAM, consulta [Using IAM roles nella IAM User Guide](#).

Dopo aver identificato il ruolo IAM, se quel ruolo ti affida la fiducia, puoi configurare il tuo SDK o lo strumento per utilizzare le autorizzazioni concesse dal ruolo. Per farlo, o. [Assumi un ruolo IAM Federazione con identità web o OpenID Connect](#)

## Assumi un ruolo IAM

Quando assume un ruolo, AWS STS restituisce un set di credenziali di sicurezza temporanee. Queste credenziali provengono da un altro profilo o dall'istanza o dal contenitore in cui è in esecuzione il codice. Altri esempi di assunzione di un ruolo includono la gestione di più account Account AWS da Amazon EC2, l'AWS CodeCommitutilizzo Account AWS di più account o l'accesso a un altro account da. AWS CodeBuild

### Fase 1: configurare un ruolo IAM

Per configurare il tuo SDK o lo strumento per assumere un ruolo, devi prima creare o identificare un ruolo specifico da assumere. I ruoli IAM vengono identificati in modo univoco utilizzando un ruolo [ARN](#). I ruoli stabiliscono relazioni di fiducia con un'altra entità, in genere all'interno dell'account o per l'accesso tra account. Per configurarlo, consulta [Creazione di ruoli IAM](#) nella Guida per l'utente IAM.

### Passaggio 2: configura l'SDK o lo strumento

Configura l'SDK o lo strumento per ottenere le credenziali da o. `credential_source`  
`source_profile`

`credential_source` Utilizzalo per ottenere credenziali da un contenitore Amazon ECS, un'istanza Amazon EC2 o da variabili di ambiente.

Utilizza `source_profile` per ottenere credenziali da un altro profilo. `source_profile` supporta anche il concatenamento dei ruoli, ossia gerarchie di profili in cui un ruolo assunto viene poi utilizzato per assumere un altro ruolo.

Quando lo specificate in un profilo, l'SDK o lo strumento effettua automaticamente la chiamata AWS STS [AssumeRole](#) API corrispondente. Per recuperare e utilizzare credenziali temporanee assumendo un ruolo, specificate i seguenti valori di configurazione nel file condiviso. AWS config Per maggiori dettagli su ciascuna di queste impostazioni, consulta la sezione. [Assumi le impostazioni del fornitore di credenziali di ruolo](#)

- `role_arn`- Dal ruolo IAM che hai creato nella fase 1
- Configura uno `source_profile` o `credential_source`
- (Facoltativo) `duration_seconds`
- (Facoltativo) `external_id`
- (Facoltativo) `mfa_serial`
- (Facoltativo) `role_session_name`

Gli esempi seguenti mostrano la configurazione di entrambe le opzioni di assunzione del ruolo in un config file condiviso:

```
role_arn = arn:aws:iam::123456789012:role/my-role-name
source_profile = profile-name-with-user-that-can-assume-role
```

```
role_arn = arn:aws:iam::123456789012:role/my-role-name
credential_source = Ec2InstanceMetadata
```

Per i dettagli su tutte le impostazioni del provider di credenziali di assunzione del ruolo, [Assumi il ruolo di fornitore di credenziali](#) consulta questa guida.

## Federazione con identità web o OpenID Connect

Quando si creano applicazioni mobili o applicazioni Web basate su client che richiedono l'accesso a AWS, AWS STS restituisce un set di credenziali di sicurezza temporanee per gli utenti federati

autenticati tramite un provider di identità pubblico (IdP). Esempi di provider di identità pubblici comprendono Login with Amazon, Facebook, Google o qualsiasi provider di identità compatibile con OpenID Connect (OIDC). Con questo metodo, gli utenti non hanno bisogno delle proprie identità o di quelle di IAM. AWS

Se utilizzi Amazon Elastic Kubernetes Service, questa funzionalità offre la possibilità di specificare diversi ruoli IAM per ciascuno dei tuoi contenitori. Kubernetes offre la possibilità di distribuire token OIDC ai contenitori, che vengono utilizzati da questo fornitore di credenziali per ottenere credenziali temporanee. Per ulteriori informazioni su questa configurazione di Amazon EKS, consulta [i ruoli IAM per gli account di servizio](#) nella Amazon EKS User Guide. Tuttavia, per un'opzione più semplice, ti consigliamo di utilizzare invece [Amazon EKS Pod Identities](#) se il tuo [SDK](#) lo supporta.

## Fase 1: configurare un provider di identità e un ruolo IAM

Per configurare la federazione con un IdP esterno, utilizza un provider di identità IAM per fornire AWS informazioni sull'IdP esterno e sulla sua configurazione. In questo modo si instaura un rapporto di fiducia tra il tuo Account AWS e l'IdP esterno. Prima di configurare l'SDK per utilizzare il token di identità web per l'autenticazione, devi prima configurare il provider di identità (IdP) e il ruolo IAM utilizzato per accedervi. Per configurarli, consulta [Creating a role for web identity o OpenID Connect Federation \(console\)](#) nella IAM User Guide.

## Passaggio 2: configura l'SDK o lo strumento

Configura l'SDK o lo strumento per utilizzare un token di identità Web AWS STS per l'autenticazione.

Quando lo specifichi in un profilo, l'SDK o lo strumento effettua automaticamente la chiamata AWS STS [AssumeRoleWithWebIdentity](#) API corrispondente per te. Per recuperare e utilizzare le credenziali temporanee utilizzando la federazione delle identità Web, specificate i seguenti valori di configurazione nel file condiviso. AWS `config` Per maggiori dettagli su ciascuna di queste impostazioni, consulta la [Assumi le impostazioni del fornitore di credenziali di ruolo](#) sezione.

- `role_arn`- Dal ruolo IAM che hai creato nella fase 1
- `web_identity_token_file`- Dall'IdP esterno
- (Facoltativo) `duration_seconds`
- (Facoltativo) `role_session_name`

Di seguito è riportato un esempio di configurazione di `config` file condivisa per assumere un ruolo con identità web:

```
[profile web-identity]  
role_arn=arn:aws:iam::123456789012:role/my-role-name  
web_identity_token_file=/path/to/a/token
```

### Note

Per le applicazioni mobili, prendi in considerazione l'utilizzo di Amazon Cognito. Amazon Cognito funge da broker di identità e svolge gran parte del lavoro federativo per te. Tuttavia, il provider di identità Amazon Cognito non è incluso nelle librerie di base degli SDK e degli strumenti come altri provider di identità. Per accedere all'API Amazon Cognito, includi il client del servizio Amazon Cognito nella build o nelle librerie del tuo SDK o strumento. Per l'utilizzo con AWS gli SDK, consulta [Esempi di codice](#) nella Amazon Cognito Developer Guide.

Per i dettagli su tutte le impostazioni del provider di credenziali di assunzione del ruolo, consulta questa guida [Assumi il ruolo di fornitore di credenziali](#).

## AWS chiavi di accesso

### Usa credenziali a breve termine

Ti consigliamo di configurare l'SDK o lo strumento da utilizzare per utilizzare [Autenticazione IAM Identity Center](#) opzioni di durata estesa della sessione.

Tuttavia, per configurare direttamente l'SDK o le credenziali temporanee dello strumento, consulta [Autenticazione utilizzando credenziali a breve termine](#)

### Usa credenziali a lungo termine

#### Warning

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

## Gestisci l'accesso in tutto Account AWS

Come best practice di sicurezza, ti consigliamo di utilizzare AWS Organizations IAM Identity Center per gestire l'accesso su tutti i tuoi Account AWS. Per ulteriori informazioni, consulta [Best Practice di sicurezza in IAM](#) nella Guida per l'utente di IAM.

Puoi creare utenti in IAM Identity Center, utilizzare Microsoft Active Directory, utilizzare un provider di identità (IdP) SAML 2.0 o federare individualmente il tuo IdP. Account AWS Utilizzando uno di questi approcci, puoi fornire un'esperienza Single Sign-On ai tuoi utenti. Puoi anche applicare l'autenticazione a più fattori (MFA) e utilizzare credenziali temporanee per l'accesso. Account AWS Ciò differisce da un utente IAM, che è una credenziale a lungo termine che può essere condivisa e che potrebbe aumentare il rischio di sicurezza per le risorse. AWS

## Crea utenti IAM solo per ambienti sandbox

Se sei un principiante AWS, potresti creare un utente IAM di prova e poi utilizzarlo per eseguire tutorial ed esplorare ciò che AWS ha da offrire. Va bene usare questo tipo di credenziale quando impari, ma ti consigliamo di evitare di utilizzarle al di fuori di un ambiente sandbox.

Per i seguenti casi d'uso, potrebbe essere utile iniziare con gli utenti IAM in: AWS

- Inizia a usare il tuo AWS SDK o il tuo strumento ed esplora Servizi AWS in un ambiente sandbox.
- L'esecuzione di script, processi e altri processi automatizzati pianificati che non supportano un processo di accesso con assistenza umana come parte del tuo apprendimento.

Se utilizzi utenti IAM al di fuori di questi casi d'uso, passa a IAM Identity Center o federa il tuo provider di identità a Account AWS il prima possibile. Per ulteriori informazioni, consulta [Identity Federation in AWS](#).

## Chiavi di accesso utente IAM sicure

È necessario ruotare regolarmente le chiavi di accesso utente IAM. Segui le indicazioni contenute nella sezione [Rotating access keys](#) nella IAM User Guide. Se ritieni di aver condiviso accidentalmente le tue chiavi di accesso utente IAM, ruota le chiavi di accesso.

Le chiavi di accesso utente IAM devono essere archiviate nel `AWS credentials` file condiviso sul computer locale. Non memorizzate le chiavi di accesso utente IAM nel codice. Non includere file di configurazione che contengono le chiavi di accesso utente IAM all'interno di alcun software di

gestione del codice sorgente. Strumenti esterni, come il progetto open source [git-secrets](#), possono aiutarti a non inserire inavvertitamente informazioni sensibili in un repository Git. Per ulteriori informazioni, consulta [IAM Identities \(users, user groups, and roles\) nella IAM User Guide](#).

Per configurare un utente IAM per iniziare, consulta [Autenticazione utilizzando credenziali a lungo termine](#).

## Autenticazione utilizzando credenziali a breve termine

Ti consigliamo di configurare l'SDK o lo strumento da utilizzare [Autenticazione IAM Identity Center](#) con opzioni di durata prolungata della sessione. Tuttavia, puoi copiare e utilizzare le credenziali temporanee disponibili nel portale di accesso. AWS Le nuove credenziali dovranno essere copiate quando scadono. È possibile utilizzare le credenziali temporanee in un profilo o utilizzarle come valori per le proprietà di sistema e le variabili di ambiente.

Configura un file di credenziali utilizzando credenziali a breve termine recuperate dal portale di accesso AWS

1. [Crea un file di credenziali condiviso](#).
2. Nel file delle credenziali, incolla il seguente testo segnaposto fino a incollare le credenziali temporanee di lavoro.

```
[default]
aws_access_key_id=<value from AWS access portal>
aws_secret_access_key=<value from AWS access portal>
aws_session_token=<value from AWS access portal>
```

3. Salvare il file. Il file `~/.aws/credentials` dovrebbe ora esistere sul tuo sistema di sviluppo locale. Questo file contiene il [profilo \[predefinito\]](#) utilizzato dall'SDK o dallo strumento se non viene specificato un profilo denominato specifico.
4. [Accedi al portale di AWS accesso](#).
5. Segui queste istruzioni per [l'aggiornamento manuale delle credenziali](#) per copiare le credenziali del ruolo IAM dal AWS portale di accesso.
  - a. Per la fase 4 delle istruzioni collegate, scegli il nome del ruolo IAM che concede l'accesso per le tue esigenze di sviluppo. Questo ruolo in genere ha un nome simile `PowerUserAccessa Developer`.
  - b. Per il passaggio 7 delle istruzioni collegate, seleziona l'opzione `Aggiungi manualmente un profilo` al file delle AWS credenziali e copia il contenuto.





- NON includete file che contengono credenziali nell'area del progetto.
- Tieni presente che tutte le credenziali memorizzate nel `AWS credentials` file condiviso vengono archiviate in testo non crittografato.

Linee guida aggiuntive per la gestione sicura delle credenziali

Per una discussione generale su come gestire in modo sicuro le credenziali AWS, consulta l'argomento contenente le [best practice per la gestione delle chiavi di accesso AWS](#) in [Riferimenti generali di AWS](#). Considera inoltre quanto segue:

- Usa [ruoli IAM](#) per le attività di Amazon Elastic Container Service (Amazon ECS).
- Usa [ruoli IAM](#) per le applicazioni in esecuzione sulle istanze Amazon EC2.

## Prerequisiti: creare un account AWS

Per utilizzare un utente IAM per accedere ai AWS servizi, sono necessari un AWS account e delle AWS credenziali.

### 1. Creazione di un account.

Per creare un AWS account, consulta la pagina [introduttiva dedicata agli utenti AWS alle prime armi](#) nella Guida di riferimento di AWS Account Management.

### 2. Creazione di un utente amministratore.

Evita di utilizzare l'account utente root (l'account iniziale creato) per accedere alla console di gestione e ai servizi. Crea invece un account utente amministratore, come illustrato in [Creazione di un utente amministratore](#) nella Guida per l'utente di IAM.

Dopo aver creato l'account utente amministratore e registrato i dettagli di accesso, assicurati di disconnetterti dall'account utente root e di accedere nuovamente utilizzando l'account amministrativo.

Nessuno di questi account è appropriato per lo sviluppo AWS o per l'esecuzione di applicazioni. AWS Come procedura ottimale, è necessario creare utenti, set di autorizzazioni o ruoli di servizio appropriati per queste attività. Per ulteriori informazioni, consulta [Applicazione delle autorizzazioni del privilegio minimo](#) nella Guida per l'utente di IAM.

## Passaggio 1: crea il tuo utente IAM

- Crea il tuo utente IAM seguendo la procedura [Creating IAM users \(console\)](#) nella IAM User Guide.
  - Per le opzioni di autorizzazione, scegli Allega direttamente le politiche in base al modo in cui desideri assegnare le autorizzazioni a questo utente.
    - La maggior parte dei tutorial SDK «Getting Started» utilizza il servizio Amazon S3 come esempio. Per fornire alla tua applicazione l'accesso completo ad Amazon S3, seleziona la `AmazonS3FullAccess` policy da allegare a questo utente.
  - Puoi ignorare i passaggi facoltativi di tale procedura.

## Fase 2: Ottieni le tue chiavi di accesso

1. Nel pannello di navigazione della console IAM, seleziona Utenti, quindi seleziona **User name** l'utente che hai creato in precedenza.
2. Nella pagina dell'utente, seleziona la pagina Credenziali di sicurezza. Quindi, in Chiavi di accesso, seleziona Crea chiave di accesso.
3. Per Creare la chiave di accesso Step 1, scegli Command Line Interface (CLI) o Codice locale. Entrambe le opzioni generano lo stesso tipo di chiave da utilizzare sia con gli SDK che con AWS CLI gli SDK.
4. Per la creazione della chiave di accesso (Fase 2), inserisci un tag opzionale e seleziona Avanti.
5. Per la fase 3 di creazione della chiave di accesso, seleziona Scarica il file.csv per salvare un `.csv` file con la chiave di accesso e la chiave di accesso segreta del tuo utente IAM. Ti serviranno queste informazioni per dopo.

### Warning

Utilizza le misure di sicurezza appropriate per proteggere queste credenziali.

6. Seleziona Done (Fatto)

## Passaggio 3: Aggiornare il file condiviso **credentials**

1. Crea o apri il AWS `credentials` file condiviso. Questo file si trova `~/.aws/credentials` su sistemi Linux e macOS e `%USERPROFILE%\aws\credentials` su Windows. Per ulteriori informazioni, consulta [Ubicazione dei file delle credenziali](#).
2. Aggiungi il seguente testo al `credentials` file condiviso. Sostituisci il valore ID di esempio e il valore della chiave di esempio con i valori del `.csv` file scaricato in precedenza.

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

3. Salvare il file.

Il `credentials` file condiviso è il modo più comune per archiviare le credenziali. Queste possono anche essere impostate come variabili di ambiente, vedi [AWS chiavi di accesso](#) per i nomi delle variabili di ambiente. Questo è un modo per iniziare, ma ti consigliamo di passare a IAM Identity Center o ad altre credenziali temporanee il prima possibile. Dopo aver abbandonato l'utilizzo di credenziali a lungo termine, ricordati di eliminare queste credenziali dal file condiviso. `credentials`

## Utilizzo dei ruoli IAM per le istanze Amazon EC2

Questo esempio illustra la configurazione di un AWS Identity and Access Management ruolo con accesso Amazon S3 da utilizzare nell'applicazione distribuita su un'istanza Amazon EC2.

Per un'istanza Amazon Elastic Compute Cloud, crea un ruolo IAM, quindi consenti alla tua istanza Amazon EC2 di accedere a quel ruolo. Per ulteriori informazioni, consulta [IAM Roles for Amazon EC2 nella Amazon EC2 User Guide for Linux Instances](#) o IAM Roles for [Amazon EC2 nella Amazon EC2 User Guide for Windows Instances](#).

### Creazione di un ruolo IAM

Crea un ruolo IAM che garantisca l'accesso in sola lettura ad Amazon S3.

1. Accedi a AWS Management Console e apri la console IAM all'indirizzo <https://console.aws.amazon.com/iam/>.
2. Nel riquadro di navigazione, seleziona Ruoli, quindi seleziona Crea ruolo.

3. Per Seleziona entità attendibile, in Tipo di entità affidabile, scegli Servizio AWS.
4. In Caso d'uso, scegli Amazon EC2, quindi seleziona Avanti.
5. Per Aggiungi autorizzazioni, seleziona la casella di controllo per Amazon S3 Read Only Access dall'elenco delle policy, quindi seleziona Avanti.
6. Inserisci un nome per il ruolo, quindi seleziona Crea ruolo. Ricorda questo nome perché ti servirà quando lancerai la tua istanza Amazon EC2.

## Avvia un'istanza Amazon EC2 e specifica il tuo ruolo IAM

Puoi avviare un'istanza Amazon EC2 con un ruolo IAM utilizzando la console Amazon EC2.

Segui le istruzioni per avviare un'istanza nella [Amazon EC2 User Guide for Linux Instances](#) o nella [Amazon EC2 User Guide for Windows Instances](#).

Quando raggiungi la pagina Review Instance Launch (Verifica del lancio dell'istanza), seleziona Edit instance details (Modifica dettagli istanza). Nel ruolo IAM, scegli il ruolo IAM che hai creato in precedenza. Completa la procedura come descritto.

### Note

È necessario creare o utilizzare un gruppo di sicurezza e una coppia di chiavi esistenti per connettersi all'istanza.

Con questa configurazione di IAM e Amazon EC2, puoi distribuire la tua applicazione sull'istanza Amazon EC2 e avrà accesso in lettura al servizio Amazon S3.

## Connect all'istanza EC2

Connect all'istanza EC2 in modo da poter trasferire l'applicazione di esempio su di essa e quindi eseguire l'applicazione. Avrai bisogno del file che contiene la parte privata della key pair che hai usato per avviare l'istanza, ovvero il file PEM.

Puoi farlo seguendo la procedura di connessione nella [Amazon EC2 User Guide for Linux Instances](#) o nella [Amazon EC2 User Guide for Windows Instances](#). Quando ti connetti, fallo in modo da poter trasferire i file dalla macchina di sviluppo all'istanza.

Se utilizzi un AWS Toolkit, spesso puoi connetterti all'istanza anche utilizzando il Toolkit. Per ulteriori informazioni, consulta la guida utente specifica per il Toolkit che utilizzi.

## Esegui l'applicazione di esempio sull'istanza EC2

1. Copia i file dell'applicazione dall'unità locale all'istanza.

Per informazioni su come trasferire file sulla tua istanza, consulta la [Amazon EC2 User Guide for Linux Instances](#) o la [Amazon EC2 User Guide for Windows Instances](#).

2. Avvia l'applicazione e verifica che funzioni con gli stessi risultati della macchina di sviluppo.
3. (Facoltativo) Verifica che l'applicazione utilizzi le credenziali fornite dal ruolo IAM.
  - a. Accedi a AWS Management Console e apri la console Amazon EC2 all'indirizzo <https://console.aws.amazon.com/ec2/>.
  - b. Seleziona l'istanza e scollega il ruolo IAM tramite Actions, Instance Settings, Attach/Replace IAM Role.
  - c. Esegui nuovamente l'applicazione e conferma che restituisca un errore di autorizzazione.

## Riferimento alle impostazioni

Gli SDK forniscono API specifiche per la lingua. Servizi AWS Si occupano di alcune delle attività più impegnative necessarie per effettuare correttamente le chiamate API, tra cui l'autenticazione, il comportamento dei tentativi di ripetizione e altro ancora. A tal fine, gli SDK dispongono di strategie flessibili per ottenere credenziali da utilizzare per le richieste, mantenere le impostazioni da utilizzare con ciascun servizio e ottenere valori da utilizzare per le impostazioni globali.

Puoi trovare informazioni dettagliate sulle impostazioni di configurazione nelle seguenti sezioni:

- [Fornitori di credenziali standardizzati](#)— Provider di credenziali comuni standardizzati su più SDK.
- [Funzionalità standardizzate](#)— Funzionalità comuni standardizzate su più SDK.

## Creazione di client di servizio

Per accedere a livello di codice Servizi AWS, gli SDK utilizzano una classe/oggetto client per ciascuno. Servizio AWS Ad esempio, se l'applicazione deve accedere ad Amazon EC2, l'applicazione crea un oggetto client Amazon EC2 per interfacciarsi con quel servizio. Quindi utilizzi il client di servizio per effettuare richieste in merito. Servizio AWS Nella maggior parte degli SDK, un oggetto client di servizio è immutabile, quindi è necessario creare un nuovo client per ogni servizio a cui si effettuano richieste e per effettuare richieste allo stesso servizio utilizzando una configurazione diversa.

## Precedenza delle impostazioni

Le impostazioni globali configurano funzionalità, fornitori di credenziali e altre funzionalità supportate dalla maggior parte degli SDK e che hanno un ampio impatto su tutti. Servizi AWS Tutti gli SDK hanno una serie di posizioni (o fonti) che controllano per trovare un valore per le impostazioni globali. Di seguito è riportata la priorità delle impostazioni di ricerca:

1. Qualsiasi impostazione esplicita impostata nel codice o su un client di servizio stesso ha la precedenza su qualsiasi altra cosa.
  - Alcune impostazioni possono essere impostate in base all'operazione e possono essere modificate secondo necessità per ogni operazione richiamata. Per l' AWS CLI operazione AWS Tools for PowerShell, queste assumono la forma di parametri specifici per operazione che vengono immessi nella riga di comando. Per un SDK, le assegnazioni esplicite possono

assumere la forma di un parametro impostato quando si crea un'istanza di un Servizio AWS client o di un oggetto di configurazione o, a volte, quando si chiama una singola API.

2. Solo Java/Kotlin: la proprietà del sistema JVM per l'impostazione è verificata. Se è impostato, quel valore viene utilizzato per configurare il client.
3. La variabile di ambiente è selezionata. Se è impostato, quel valore viene utilizzato per configurare il client.
4. L'SDK verifica l'impostazione nel `credentials` file condiviso. Se è impostato, il client lo utilizza.
5. Il `config` file condiviso per l'impostazione. Se l'impostazione è presente, l'SDK la utilizza.
  - La variabile di `AWS_PROFILE` ambiente o la proprietà di sistema `aws.profile` JVM possono essere utilizzate per specificare il profilo caricato dall'SDK.
6. Qualsiasi valore predefinito fornito dal codice sorgente SDK stesso viene utilizzato per ultimo.

#### Note

Alcuni SDK e strumenti potrebbero eseguire il check-in in un ordine diverso. Inoltre, alcuni SDK e strumenti supportano altri metodi di archiviazione e recupero dei parametri. [Ad esempio, AWS SDK for .NET supporta una fonte aggiuntiva chiamata SDK Store.](#) Per ulteriori informazioni sui provider esclusivi di un SDK o di uno strumento, consulta la guida specifica per l'SDK o lo strumento che stai utilizzando.

L'ordine determina quali metodi hanno la precedenza e sostituiscono gli altri. Ad esempio, se configuri un profilo nel `config` file condiviso, questo viene trovato e utilizzato solo dopo che l'SDK o lo strumento hanno prima verificato le altre posizioni. Ciò significa che se inserisci un'impostazione nel `credentials` file, questa viene utilizzata al posto di quella trovata nel `config` file. Se si configura una variabile di ambiente con un'impostazione e un valore, questa avrà la precedenza su tale impostazione `credentials` sia nei `config` file che. Infine, un'impostazione sulla singola operazione (parametro della AWS CLI riga di comando o parametro API) o nel codice sovrascriverebbe tutti gli altri valori di quell'unico comando.

## Configelencò delle impostazioni dei file

Le impostazioni elencate nella tabella seguente possono essere assegnate nel AWS `config` file condiviso. Sono globali e riguardano tutti Servizi AWS.

Nome dell'impostazione	Informazioni
api_versions	<a href="#">Impostazioni generali di configurazione</a>
aws_access_key_id	<a href="#">AWS chiavi di accesso</a>
aws_secret_access_key	<a href="#">AWS chiavi di accesso</a>
aws_session_token	<a href="#">AWS chiavi di accesso</a>
ca_bundle	<a href="#">impostazioni generali di configurazione</a>
credential_process	<a href="#">Provider di credenziali di processo</a>
credential_source	<a href="#">Assumi il ruolo di fornitore di credenziali</a>
defaults_mode	<a href="#">Impostazioni predefinite di configurazione intelligente</a>
disable_request_compression	<a href="#">Richiedi la compressione</a>
duration_seconds	<a href="#">Assumi il ruolo di fornitore di credenziali</a>
ec2_metadata_service_endpoint	<a href="#">Fornitore di credenziali IMDS</a>
ec2_metadata_service	<a href="#">Fornitore di credenziali IMDS</a>



Nome dell'impostazione	Informazioni
ce_endpoint_mode	
ec2_metadata_v1_disabled	<a href="#">Fornitore di credenziali IMDS</a>
endpoint_discovery_enabled	<a href="#">Individuazione degli endpoint</a>
endpoint_url	<a href="#">Endpoint specifici del servizio</a>
external_id	<a href="#">Assumi il ruolo di fornitore di credenziali</a>
ignore_configured_endpoint_urls	<a href="#">Endpoint specifici del servizio</a>
max_attempts	<a href="#">Comportamento dei nuovi tentativi</a>
metadata_service_num_attempts	<a href="#">Metadati delle istanze Amazon EC2</a>
metadata_service_timeout	<a href="#">Metadati delle istanze Amazon EC2</a>
mfa_serial	<a href="#">Assumi il ruolo di fornitore di credenziali</a>
output	<a href="#">Impostazioni generali di configurazione</a>
parameter_validation	<a href="#">Impostazioni generali di configurazione</a>
region	<a href="#">Regione AWS</a>

Nome dell'impostazione	Informazioni	
<code>request_max_in_compression_size_bytes</code>	<a href="#">Richiedi la compressione</a>	
<code>retry_mode</code>	<a href="#">Comportamento dei nuovi tentativi</a>	
<code>role_arn</code>	<a href="#">Assumi il ruolo di fornitore di credenziali</a>	
<code>role_session_name</code>	<a href="#">Assumi il ruolo di fornitore di credenziali</a>	
<code>s3_disable_multiregion_access_points</code>	<a href="#">Punti di accesso multi-Regione di Amazon S3</a>	
<code>s3_use_arn_region</code>	<a href="#">Access point Amazon S3</a>	
<code>source_profile</code>	<a href="#">Assumi il ruolo di fornitore di credenziali</a>	
<code>sso_account_id</code>	<a href="#">Provider di credenziali IAM Identity Center</a>	
<code>sso_region</code>	<a href="#">Provider di credenziali IAM Identity Center</a>	
<code>sso_registration_scopes</code>	<a href="#">Provider di credenziali IAM Identity Center</a>	
<code>sso_role_name</code>	<a href="#">Provider di credenziali IAM Identity Center</a>	
<code>sso_start_url</code>	<a href="#">Provider di credenziali IAM Identity Center</a>	
<code>sts_regional_endpoints</code>	<a href="#">AWS STS Endpoint regionalizzati</a>	

Nome dell'impostazione	Informazioni
<code>use_dualstack_endpoint</code>	<a href="#">Endpoint dual-stack e FIPS</a>
<code>use_fips_endpoint</code>	<a href="#">Endpoint dual-stack e FIPS</a>
<code>web_identity_token_file</code>	<a href="#">Assumi il ruolo di fornitore di credenziali</a>

## Credentialseleco delle impostazioni dei file

Le impostazioni elencate nella tabella seguente possono essere assegnate nel AWS `credentials` file condiviso. Sono globali e riguardano tutti Servizi AWS.

Nome dell'impostazione	Informazioni
<code>aws_access_key_id</code>	<a href="#">AWS chiavi di accesso</a>
<code>aws_secret_access_key</code>	<a href="#">AWS chiavi di accesso</a>
<code>aws_session_token</code>	<a href="#">AWS chiavi di accesso</a>

## elenco delle variabili di ambiente

Le variabili di ambiente supportate dalla maggior parte degli SDK sono elencate nella tabella seguente. Sono globali e riguardano tutti Servizi AWS.

Nome dell'impostazione	Informazioni
AWS_ACCESS_KEY_ID	<a href="#">AWS chiavi di accesso</a>
AWS_CA_BUNDLE	<a href="#">impostazioni generali di configurazione</a>
AWS_CONFIG_FILE	<a href="#">Ubicazione degli elementi condivisi config e credentials dei file</a>
AWS_CONTAINER_AUTHORIZATION_TOKEN	<a href="#">Fornitore di credenziali per contenitori</a>
AWS_CONTAINER_AUTHORIZATION_TOKEN_FILE	<a href="#">Fornitore di credenziali per contenitori</a>
AWS_CONTAINER_CREDENTIALS_FULL_URI	<a href="#">Fornitore di credenziali per contenitori</a>
AWS_CONTAINER_CREDENTIALS_RELATIVE_URI	<a href="#">Fornitore di credenziali per contenitori</a>
AWS_DEFAULTS_MODE	<a href="#">Impostazioni predefinite di configurazione intelligenti</a>
AWS_DISABLE_REQUEST_COMPRESSION	<a href="#">Richiedi la compressione</a>

Nome dell'impostazione	Informazioni
AWS_EC2_METADATA_DISABLED	<a href="#">Fornitore di credenziali IMDS</a>
AWS_EC2_METADATA_SERVICE_ENDPOINT	<a href="#">Fornitore di credenziali IMDS</a>
AWS_EC2_METADATA_SERVICE_ENDPOINT_MODE	<a href="#">Fornitore di credenziali IMDS</a>
AWS_EC2_METADATA_V1_DISABLED	<a href="#">Fornitore di credenziali IMDS</a>
AWS_ENABLE_ENDPOINT_DISCOVERY	<a href="#">Individuazione degli endpoint</a>
AWS_ENDPOINT_URL	<a href="#">Endpoint specifici del servizio</a>
AWS_ENDPOINT_URL_<SERVICE>	<a href="#">Endpoint specifici del servizio</a>
AWS_IAM_ROLE_ARN	<a href="#">Assumi il ruolo di fornitore di credenziali</a>
AWS_IAM_ROLE_SESSION_NAME	<a href="#">Assumi il ruolo di fornitore di credenziali</a>

Nome dell'impostazione	Informazioni
AWS_IGNORE_CONFIGURED_ENDPOINT_URLS	<a href="#">Endpoint specifici del servizio</a>
AWS_MAX_ATTEMPTS	<a href="#">Comportamento dei nuovi tentativi</a>
AWS_METADATA_SERVICE_NUM_ATTEMPTS	<a href="#">Metadati delle istanze Amazon EC2</a>
AWS_METADATA_SERVICE_TIMEOUT	<a href="#">Metadati delle istanze Amazon EC2</a>
AWS_PROFILE	<a href="#">Condivisi config e file credentials</a>
AWS_REGION	<a href="#">Regione AWS</a>
AWS_REQUEST_MIN_COMPRESSION_SIZE_BYTES	<a href="#">Richiedi la compressione</a>
AWS_RETRY_MODE	<a href="#">Comportamento dei nuovi tentativi</a>
AWS_S3_DISABLE_MULTIREGION_ACCESS_POINTS	<a href="#">Punti di accesso multi-Regione di Amazon S3</a>
AWS_S3_USE_ARN_REGION	<a href="#">Access point Amazon S3</a>

Nome dell'impostazione	Informazioni
AWS_SECRET_ACCESS_KEY	<a href="#">AWS chiavi di accesso</a>
AWS_SESSION_TOKEN	<a href="#">AWS chiavi di accesso</a>
AWS_SHARED_CREDENTIALS_FILE	<a href="#">Ubicazione dei credentials file condivisi config e</a>
AWS_STS_REGIONAL_ENDPOINTS	<a href="#">AWS STS Endpoint regionalizzati</a>
AWS_USE_DUALSTACK_ENDPOINT	<a href="#">Endpoint dual-stack e FIPS</a>
AWS_USE_FIPS_ENDPOINT	<a href="#">Endpoint dual-stack e FIPS</a>
AWS_WEB_IDENTITY_TOKEN_FILE	<a href="#">Assumi il ruolo di fornitore di credenziali</a>

## Elenco delle proprietà del sistema JVM

È possibile utilizzare le seguenti proprietà del sistema JVM per AWS SDK for Java e SDK AWS for Kotlin (destinate alla JVM). [the section called “Come impostare le proprietà del sistema JVM”](#) Per istruzioni su come impostare le proprietà del sistema JVM, vedere.

Nome dell'impostazione	Informazioni
<code>aws.accessKeyId</code>	<a href="#">AWS chiavi di accesso</a>
<code>aws.configFile</code>	<a href="#">Ubicazione dei credentials file condivisi config e</a>
<code>aws.defaultsMode</code>	<a href="#">Impostazioni predefinite di configurazione intelligente</a>
<code>aws.disableEc2MetadataV1</code>	<a href="#">Provider di credenziali IMDS</a>
<code>aws.disableRequestCompression</code>	<a href="#">Richiedi la compressione</a>
<code>aws.ec2MetadataServiceEndpoint</code>	<a href="#">Fornitore di credenziali IMDS</a>
<code>aws.ec2MetadataServiceEndpointMode</code>	<a href="#">Fornitore di credenziali IMDS</a>
<code>aws.endpointDiscoveryEnabled</code>	<a href="#">Individuazione degli endpoint</a>
<code>aws.endpointUrl</code>	<a href="#">Endpoint specifici del servizio</a>
<code>aws.endpointUrl&lt;ServiceName&gt;</code>	<a href="#">Endpoint specifici del servizio</a>



Nome dell'impostazione	Informazioni
<code>aws.ignoreConfiguredEndpointUrls</code>	<a href="#">Endpoint specifici del servizio</a>
<code>aws.maxAttempts</code>	<a href="#">Comportamento dei nuovi tentativi</a>
<code>aws.profile</code>	<a href="#">Condivisi config e file credentials</a>
<code>aws.region</code>	<a href="#">Regione AWS</a>
<code>aws.requestMinCompressionSizeBytes</code>	<a href="#">Richiedi la compressione</a>
<code>aws.retryMode</code>	<a href="#">Comportamento dei nuovi tentativi</a>
<code>aws.roleArn</code>	<a href="#">Assumi il ruolo di fornitore di credenziali</a>
<code>aws.roleSessionName</code>	<a href="#">Assumi il ruolo di fornitore di credenziali</a>
<code>aws.s3DisableMultiRegionAccessPoints</code>	<a href="#">Punti di accesso multi-Regione di Amazon S3</a>
<code>aws.s3UseArnRegion</code>	<a href="#">Access point Amazon S3</a>
<code>aws.secretAccessKey</code>	<a href="#">AWS chiavi di accesso</a>
<code>aws.sessionToken</code>	<a href="#">AWS chiavi di accesso</a>

Nome dell'impostazione	Informazioni
<code>aws.shareCredentialsFile</code>	<a href="#">Ubicazione dei credentials file condivisi</a> e <a href="#">config</a>
<code>aws.useDualStackEndpoint</code>	<a href="#">Endpoint dual-stack e FIPS</a>
<code>aws.useFipsEndpoint</code>	<a href="#">Endpoint dual-stack e FIPS</a>
<code>aws.webIdentityTokenFile</code>	<a href="#">Assumi il ruolo di fornitore di credenziali</a>

## Fornitori di credenziali standardizzati

Molti fornitori di credenziali sono stati standardizzati in base a impostazioni predefinite coerenti e funzionano allo stesso modo su molti SDK. Questa coerenza aumenta la produttività e la chiarezza durante la codifica su più SDK. Tutte le impostazioni possono essere sovrascritte nel codice. Per i dettagli, consulta la tua API SDK specifica.

### Important

Non tutti gli SDK supportano tutti i provider e nemmeno tutti gli aspetti all'interno di un provider.

### Argomenti

- [Catena di fornitori di credenziali](#)
- [AWS chiavi di accesso](#)
- [Assumi il ruolo di fornitore di credenziali](#)
- [Provider di credenziali per container](#)

- [Provider di credenziali IAM Identity Center](#)
- [Provider di credenziali IMDS](#)
- [Provider di credenziali di processo](#)

## Catena di fornitori di credenziali

Tutti gli SDK dispongono di una serie di posizioni (o fonti) che controllano per trovare credenziali valide da utilizzare per effettuare una richiesta a un Servizio AWS. Dopo aver trovato credenziali valide, la ricerca viene interrotta. Questa ricerca sistematica è chiamata catena di fornitori di credenziali predefinita.

Sebbene la catena distinta utilizzata da ciascun SDK sia diversa, il più delle volte include fonti come le seguenti:

Fornitore di credenziali	Description
<a href="#">AWS chiavi di accesso</a>	AWS chiavi di accesso per un utente IAM (ad esempio <code>AWS_ACCESS_KEY_ID</code> , and <code>AWS_SECRET_ACCESS_KEY</code> ).
<a href="#">Federazione con identità web o OpenID Connect</a> - Assumi il ruolo di fornitore di credenziali	Accedi utilizzando un provider di identità (IdP) esterno noto, come Login with Amazon, Facebook, Google o qualsiasi altro IdP compatibile con OpenID Connect (OIDC). Assumi le autorizzazioni di un ruolo IAM utilizzando un token di identità web di (). AWS Security Token Service AWS STS
<a href="#">Provider di credenziali IAM Identity Center</a>	Ottieni credenziali da. AWS IAM Identity Center
<a href="#">Assumi il ruolo di fornitore di credenziali</a>	Ottieni l'accesso ad altre risorse assumendo le autorizzazioni di un ruolo IAM. (Recupera e usa le credenziali temporanee per un ruolo).
<a href="#">Provider di credenziali per container</a>	Credenziali Amazon Elastic Container Service (Amazon ECS) e Amazon Elastic Kubernetes Service (Amazon

Fornitore di credenziali	Description
	EKS). Il provider di credenziali del contenitore recupera le credenziali per l'applicazione containerizzata del cliente.
<a href="#">Provider di credenziali di processo</a>	Fornitore di credenziali personalizzate. Ottieni le tue credenziali da una fonte o da un processo esterno, incluso IAM Roles Anywhere.
<a href="#">Provider di credenziali IMDS</a>	Credenziali del profilo dell'istanza Amazon Elastic Compute Cloud (Amazon EC2). Associa un ruolo IAM a ciascuna delle tue istanze EC2. Le credenziali temporanee e per quel ruolo vengono rese disponibili per il codice in esecuzione nell'istanza. Le credenziali vengono fornite tramite il servizio di metadati Amazon EC2.

Per ogni fase della catena, esistono diversi modi per assegnare i valori di impostazione. L'impostazione dei valori specificati nel codice ha sempre la precedenza. Tuttavia, ci sono anche [Variabili di ambiente](#) e il [Condivisi config e credentials file](#). Per ulteriori informazioni, consulta [Precedenza delle impostazioni](#).

## AWS chiavi di accesso

### Warning

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

AWS le chiavi di accesso per un utente IAM possono essere utilizzate come AWS credenziali. L'AWS SDK utilizza automaticamente queste AWS credenziali per firmare le richieste API AWS, in modo che i carichi di lavoro possano accedere alle AWS risorse e ai dati in modo sicuro e conveniente. Si consiglia di utilizzare sempre il `aws_session_token` modo che le credenziali siano temporanee e non più valide dopo la scadenza. L'utilizzo di credenziali a lungo termine non è consigliato.

**Note**

Se non AWS è possibile aggiornare queste credenziali temporanee, è AWS possibile estenderne la validità in modo da non influire sui carichi di lavoro.

Il `AWS credentials` file condiviso è la posizione consigliata per l'archiviazione delle informazioni sulle credenziali perché si trova in modo sicuro all'esterno delle directory di origine dell'applicazione e separato dalle impostazioni specifiche dell'SDK del file condiviso. `config`

Per ulteriori informazioni sulle AWS credenziali e sull'utilizzo delle chiavi di accesso, consulta le [credenziali di AWS sicurezza](#) e la [gestione delle chiavi di accesso per gli utenti IAM nella Guida per l'utente IAM](#).

Configura questa funzionalità utilizzando quanto segue:

**aws\_access\_key\_id**- impostazione dei AWS `config` file condivisi, **aws\_access\_key\_id**- impostazione condivisa dei AWS `credentials` file (metodo consigliato), **AWS\_ACCESS\_KEY\_ID**- variabile d'ambiente, **aws.accessKeyId**- Proprietà del sistema JVM: solo Java/Kotlin

Specifica la chiave di AWS accesso utilizzata come parte delle credenziali per autenticare l'utente.

**aws\_secret\_access\_key**- impostazione dei file condivisi AWS `config`, **aws\_secret\_access\_key**- impostazione condivisa dei AWS `credentials` file (metodo consigliato), **AWS\_SECRET\_ACCESS\_KEY**- variabile d'ambiente, **aws.secretAccessKey**- Proprietà del sistema JVM: solo Java/Kotlin

Specifica la chiave AWS segreta utilizzata come parte delle credenziali per autenticare l'utente.

**aws\_session\_token**- impostazione dei file condivisi AWS `config`, **aws\_session\_token**- impostazione condivisa dei AWS `credentials` file (metodo consigliato), **AWS\_SESSION\_TOKEN**- variabile d'ambiente, **aws.sessionToken**- Proprietà del sistema JVM: solo Java/Kotlin

Specifica un token di AWS sessione utilizzato come parte delle credenziali per autenticare l'utente. Questo valore viene ricevuto come parte delle credenziali temporanee restituite dalle richieste di assunzione di un ruolo riuscite. È richiesto un token di sessione solo se si specificano manualmente credenziali di sicurezza temporanee. Tuttavia, ti consigliamo di utilizzare sempre credenziali di sicurezza temporanee anziché credenziali a lungo termine. Per consigli sulla sicurezza, consulta [Best practice di sicurezza in IAM](#).

Per istruzioni su come ottenere questi valori, consulta [Autenticazione utilizzando credenziali a breve termine](#).

Esempio di impostazione di questi valori obbligatori nel `credentials` file config or:

```
[default]
aws_access_key_id = AKIAIOSFODNN7EXAMPLE
aws_secret_access_key = wJalrXUtnFEMI/K7MDENG/bPxrFcYEXAMPLEKEY
aws_session_token = AQoEXAMPLEH4aoAH0gNCAPy...truncated...zrkuWJ0gQs8IZZaIv2BXIa2R40lgk
```

Esempio in Linux/macOS di impostazione delle variabili di ambiente tramite riga di comando:

```
export AWS_ACCESS_KEY_ID=AKIAIOSFODNN7EXAMPLE
export AWS_SECRET_ACCESS_KEY=wJalrXUtnFEMI/K7MDENG/bPxrFcYEXAMPLEKEY
export
  AWS_SESSION_TOKEN=AQoEXAMPLEH4aoAH0gNCAPy...truncated...zrkuWJ0gQs8IZZaIv2BXIa2R40lgk
```

Esempio in Windows di impostazione delle variabili di ambiente tramite riga di comando:

```
setx AWS_ACCESS_KEY_ID AKIAIOSFODNN7EXAMPLE
setx AWS_SECRET_ACCESS_KEY wJalrXUtnFEMI/K7MDENG/bPxrFcYEXAMPLEKEY
setx
  AWS_SESSION_TOKEN AQoEXAMPLEH4aoAH0gNCAPy...truncated...zrkuWJ0gQs8IZZaIv2BXIa2R40lgk
```

## Compatibilità con gli AWS SDK

I seguenti SDK supportano le funzionalità e le impostazioni descritte in questo argomento. Vengono annotate eventuali eccezioni parziali. Tutte le impostazioni delle proprietà del sistema JVM sono supportate solo da AWS SDK for Java and the. SDK AWS for Kotlin

SDK	Sì	Note o ulteriori informazioni
<a href="#">AWS CLI v2</a>	Sì	
<a href="#">SDK per C++</a>	Sì	configfile condiviso non supportato.
<a href="#">SDK per Go V2 (1.x)</a>	Sì	

SDK	Sì	Note o ulteriori informazioni
<a href="#">SDK per Go 1.x (V1)</a>	Sì	<a href="#">Per utilizzare le impostazioni dei config file condivisi, devi attivare il caricamento dal file di configurazione; vedi Sessioni.</a>
<a href="#">SDK per Java 2.x</a>	Sì	
<a href="#">SDK per Java 1.x</a>	Sì	
<a href="#">SDK per 3.x JavaScript</a>	Sì	
<a href="#">SDK per 2.x JavaScript</a>	Sì	
<a href="#">SDK per Kotlin</a>	Sì	
<a href="#">SDK per .NET 3.x</a>	Sì	Variabili di ambiente non supportate.
<a href="#">SDK per PHP 3.x</a>	Sì	
<a href="#">SDK per Python (Boto3)</a>	Sì	
<a href="#">SDK per Ruby 3.x</a>	Sì	
<a href="#">SDK per Rust</a>	Sì	
<a href="#">Strumenti per PowerShell</a>	Sì	Variabili di ambiente non supportate.

## Assumi il ruolo di fornitore di credenziali

Assumere un ruolo implica l'utilizzo di un set di credenziali di sicurezza temporanee per accedere a AWS risorse a cui altrimenti non avresti accesso. Le credenziali temporanee sono costituite da un ID chiave di accesso, una chiave di accesso segreta e un token di sicurezza.

Per configurare l'SDK o lo strumento per assumere un ruolo, devi prima creare o identificare un ruolo specifico da assumere. I ruoli IAM sono identificati in modo univoco da un ruolo Amazon Resource Name ([ARN](#)). I ruoli stabiliscono relazioni di fiducia con un'altra entità. L'entità affidabile che utilizza il ruolo potrebbe essere un'altra Servizio AWS Account AWS, un provider di identità Web o una federazione OIDC o SAML.

Dopo aver identificato il ruolo IAM, se quel ruolo ti affida la fiducia, puoi configurare il tuo SDK o lo strumento per utilizzare le autorizzazioni concesse dal ruolo. A tale scopo, utilizza le seguenti impostazioni.

Per indicazioni su come iniziare a utilizzare queste impostazioni, [Assunzione di un ruolo](#) consulta questa guida.

## Assumi le impostazioni del fornitore di credenziali di ruolo

Configura questa funzionalità utilizzando quanto segue:

### **credential\_source**- impostazione dei file condivisi AWS **config**

Utilizzato all'interno delle istanze Amazon EC2 o dei contenitori Amazon Elastic Container Service per specificare dove l'SDK o lo strumento possono trovare le credenziali autorizzate ad assumere il ruolo specificato con il parametro. `role_arn`

Valore predefinito: Nessuno

Valori validi:

- **Ambiente:** [specifica che l'SDK o lo strumento devono recuperare le credenziali di origine dalle variabili di ambiente e. `AWS\_ACCESS\_KEY\_ID` `AWS\_SECRET\_ACCESS\_KEY`](#)
- **Ec2 InstanceMetadata:** specifica che l'SDK o lo strumento deve utilizzare il [ruolo IAM collegato al profilo dell'istanza EC2 per ottenere le](#) credenziali di origine.
- **EcsContainer**— Specifica che l'SDK o lo strumento deve utilizzare il [ruolo IAM collegato al contenitore ECS per ottenere le credenziali di origine.](#)

Non è possibile specificare sia `credential_source` sia `source_profile` nello stesso profilo.

Esempio di impostazione in un `config` file per indicare che le credenziali devono provenire da Amazon EC2:

```
credential_source = Ec2InstanceMetadata
role_arn = arn:aws:iam::123456789012:role/my-role-name
```

### **duration\_seconds**- impostazione di file condivisi AWS **config**

Specifica la durata massima della sessione del ruolo, in secondi.

Questa impostazione si applica solo quando il profilo specifica di assumere un ruolo.

Valore predefinito: 3600 secondi (un'ora)



Valori validi: il valore può variare da 900 secondi (15 minuti) fino all'impostazione della durata massima della sessione configurata per il ruolo (che può essere un massimo di 43200 secondi o 12 ore). Per ulteriori informazioni, consulta [Visualizza l'impostazione della durata massima della sessione per un ruolo](#) nella Guida per l'utente IAM.

Esempio di impostazione in un config file:

```
duration_seconds = 43200
```

### **external\_id**- impostazione di AWS **config** file condivisi

Specifica un identificatore univoco che viene utilizzato da terze parti per assumere un ruolo negli account dei relativi clienti.

Questa impostazione si applica solo quando il profilo specifica di assumere un ruolo e la politica di fiducia per il ruolo richiede un valore perExternalId. Il valore è mappato al ExternalId parametro che viene passato all'AssumeRoleoperazione quando il profilo specifica un ruolo.

Valore predefinito: Nessuno.

Valori validi: vedi [Come utilizzare un ID esterno per concedere l'accesso alle tue AWS risorse a una terza parte](#) nella Guida per l'utente IAM.

Esempio di impostazione in un config file:

```
external_id = unique_value_assigned_by_3rd_party
```

### **mfa\_serial**- impostazione di AWS **config** file condivisi

Specifica l'identificazione o il numero di serie di un dispositivo di autenticazione a più fattori (MFA) che l'utente deve utilizzare quando assume un ruolo.

Richiesto quando si assume un ruolo in cui la politica di attendibilità per quel ruolo include una condizione che richiede l'autenticazione MFA.

Valore predefinito: Nessuno.

Valori validi: il valore può essere un numero di serie per un dispositivo hardware (ad esempio GAHT12345678) o un Amazon Resource Name (ARN) per un dispositivo MFA virtuale. Per ulteriori informazioni sull'MFA, consulta [Configurazione dell'accesso all'API protetto da MFA nella Guida per l'utente IAM](#).

## Esempio di impostazione in un file: `config`

```
mfa_serial = arn:aws:iam::123456789012:mfa/my-user-name
```

**role\_arn**- impostazione di AWS `config` file condivisi, **AWS\_IAM\_ROLE\_ARN**- variabile d'ambiente, **aws.roleArn**- Proprietà del sistema JVM: solo Java/Kotlin

Specifica l'Amazon Resource Name (ARN) di un ruolo IAM che desideri utilizzare per eseguire le operazioni richieste utilizzando questo profilo.

Valore predefinito: Nessuno.

Valori validi: il valore deve essere l'ARN di un ruolo IAM, formattato come segue:

```
arn:aws:iam::account-id:role/role-name
```

Inoltre, è necessario specificare anche una delle seguenti impostazioni:

- **source\_profile**— Per identificare un altro profilo da utilizzare per trovare le credenziali autorizzate ad assumere il ruolo in questo profilo.
- **credential\_source**— Utilizzare credenziali identificate dalle variabili di ambiente correnti o credenziali allegate a un profilo di istanza Amazon EC2 o a un'istanza di container Amazon ECS.
- **web\_identity\_token\_file**— Utilizzare provider di identità pubblici o qualsiasi provider di identità compatibile con OpenID Connect (OIDC) per gli utenti che sono stati autenticati in un'applicazione mobile o web.

**role\_session\_name**- impostazione di file condivisi AWS `config`, **AWS\_IAM\_ROLE\_SESSION\_NAME**- variabile d'ambiente, **aws.roleSessionName**- Proprietà del sistema JVM: solo Java/Kotlin

Specifica il nome da associare alla sessione del ruolo. Questo nome appare nei AWS CloudTrail registri delle voci associate a questa sessione, il che può essere utile durante il controllo.

Valore predefinito: un parametro opzionale. Se non fornisci questo valore, viene generato automaticamente un nome di sessione se il profilo assume un ruolo.

Valori validi: forniti al `RoleSessionName` parametro quando l' AWS API AWS CLI o l'API richiama l'`AssumeRole` operazione (o operazioni come l'`AssumeRoleWithWebIdentity` operazione) per tuo conto. Il valore diventa parte dell'utente presunto Amazon Resource Name (ARN) che puoi interrogare e viene visualizzato come parte delle voci di CloudTrail registro per le operazioni richiamate da questo profilo.

```
arn:aws:sts::123456789012:assumed-role/my-role-name/my-role_session_name.
```

Esempio di impostazione di questo valore in un config file:

```
role_session_name = my-role-session-name
```

### **source\_profile**- impostazione di AWS **config** file condivisi

Specifica un altro profilo le cui credenziali vengono utilizzate per assumere il ruolo specificato dall'`role_arn` impostazione nel profilo originale. Per informazioni su come vengono utilizzati i profili negli archivi condivisi AWS config e nei credentials file, consulta. [Condivisi config e credentials file](#)

Se si specifica un profilo che è anche un profilo di assunzione, ogni ruolo verrà assunto in ordine sequenziale per risolvere completamente le credenziali. Questa catena viene interrotta quando l'SDK incontra un profilo con credenziali. Il concatenamento dei ruoli limita la sessione di ruolo AWS CLI o dell' AWS API a un massimo di un'ora e non può essere aumentato. Per ulteriori informazioni, consulta [i termini e i concetti relativi ai ruoli](#) nella Guida per l'utente IAM.

Valore predefinito: Nessuno.

Valori validi: una stringa di testo costituita dal nome di un profilo definito nei credentials file config and. È inoltre necessario specificare un valore per `role_arn` nel profilo corrente.

Non è possibile specificare sia `credential_source` sia `source_profile` nello stesso profilo.

Esempio di impostazione in un file di configurazione:

```
[profile A]
source_profile = B
role_arn = arn:aws:iam::123456789012:role/RoleA

[profile B]
aws_access_key_id=AKIAIOSFODNN7EXAMPLE
aws_secret_access_key=wJalrXUtnFEMI/K7MDENG/bPxrRfiCYEXAMPLEKEY
aws_session_token=IQoJb3JpZ2LuX2IQoJb3JpZ2LuX2IQoJb3JpZ2LuX2IQoJb3JpZ2LuX2IQoJb3JpZVERYLONGS
```

Nell'esempio precedente il profilo A utilizza le credenziali del profilo B. Quando specificate che l'SDK o lo strumento devono utilizzare il A profilo, l'SDK o lo strumento cerca automaticamente le credenziali per il B profilo collegato e le utilizza per richiedere credenziali temporanee per il ruolo

IAM specificato. L'SDK o lo strumento utilizza l'operazione [sts](#): in background per AssumeRole eseguire questa operazione. Queste credenziali temporanee vengono quindi utilizzate dal codice per accedere alle risorse. AWS Al ruolo specificato devono essere associate politiche di autorizzazione IAM che consentano l'esecuzione del codice richiesto, ad esempio il comando o il Servizio AWS metodo API.

**web\_identity\_token\_file**- impostazione condivisa AWS **config** dei file,

**AWS\_WEB\_IDENTITY\_TOKEN\_FILE**- variabile d'ambiente, **aws.webIdentityTokenFile**-

Proprietà del sistema JVM: solo Java/Kotlin

Specifica il percorso di un file che contiene un token di accesso da un provider [OAuth 2.0 supportato](#) o da un provider di identità [OpenID Connect ID](#).

Questa impostazione consente l'autenticazione utilizzando provider di federazione delle identità Web, come [Google](#), [Facebook](#) e [Amazon](#), tra molti altri. L'SDK o lo strumento di sviluppo carica il contenuto di questo file e lo passa come WebIdentityToken argomento quando chiama l'AssumeRoleWithWebIdentityoperazione per conto dell'utente.

Valore predefinito: Nessuno.

Valori validi: questo valore deve essere un percorso e un nome di file. Il file deve contenere un token di accesso OAuth 2.0 o un token OpenID Connect fornito da un provider di identità. I percorsi relativi vengono trattati come relativi alla directory di lavoro del processo.

## Compatibilità con gli AWS SDK

I seguenti SDK supportano le funzionalità e le impostazioni descritte in questo argomento. Vengono annotate eventuali eccezioni parziali. Tutte le impostazioni delle proprietà del sistema JVM sono supportate solo da AWS SDK for Java and the. SDK AWS for Kotlin

SDK	Sì o	Note o ulteriori informazioni
<a href="#">AWS CLI v2</a>	Sì	
<a href="#">SDK per C++</a>	Parziale	credential_source non supportato. duration_seconds non supportato. mfa_serial non supportato.
<a href="#">SDK per Go V2 (1.x)</a>	Sì	

SDK	Sì	Note o ulteriori informazioni
<a href="#">SDK per Go 1.x (V1)</a>	Sì	<a href="#">Per utilizzare le impostazioni dei config file condivisi, devi attivare il caricamento dal file di configurazione; vedi Sessioni.</a>
<a href="#">SDK per Java 2.x</a>	Parziale	<code>mfa_serial</code> non supportato. Usa <code>AWS_ROLE_ARN</code> al posto di <code>AWS_IAM_ROLE_ARN</code> . Utilizzare <code>AWS_ROLE_SESSION_NAME</code> al posto di <code>AWS_IAM_ROLE_SESSION_NAME</code> .
<a href="#">SDK per Java 1.x</a>	Parziale	<code>mfa_serial</code> non supportato.
<a href="#">SDK per 3.x JavaScript</a>	Sì	
<a href="#">SDK per 2.x JavaScript</a>	Parziale	<code>credential_source</code> non supportato.
<a href="#">SDK per Kotlin</a>	Sì	Usa invece di <code>AWS_ROLE_ARN</code> . <code>AWS_IAM_ROLE_ARN</code> . Utilizzare <code>AWS_ROLE_SESSION_NAME</code> al posto di <code>AWS_IAM_ROLE_SESSION_NAME</code> .
<a href="#">SDK per .NET 3.x</a>	Sì	
<a href="#">SDK per PHP 3.x</a>	Sì	
<a href="#">SDK per Python (Boto3)</a>	Sì	
<a href="#">SDK per Ruby 3.x</a>	Sì	
<a href="#">SDK per Rust</a>	Sì	
<a href="#">Strumenti per PowerShell</a>	Sì	

## Provider di credenziali per container

Il provider di credenziali del contenitore recupera le credenziali per l'applicazione containerizzata del cliente. Questo provider di credenziali è utile per i clienti di Amazon Elastic Container Service (Amazon ECS) e Amazon Elastic Kubernetes Service (Amazon EKS). Gli SDK tentano di caricare le credenziali dall'endpoint HTTP specificato tramite una richiesta GET.

Se utilizzi Amazon ECS, ti consigliamo di utilizzare un task IAM Role per migliorare l'isolamento, l'autorizzazione e la verificabilità delle credenziali. Una volta configurato, Amazon ECS imposta la variabile di ambiente `AWS_CONTAINER_CREDENTIALS_RELATIVE_URI` utilizzata dagli SDK e dagli strumenti per ottenere le credenziali. Per configurare Amazon ECS per questa funzionalità, consulta il [ruolo Task IAM](#) nella Amazon Elastic Container Service Developer Guide.

Se usi Amazon EKS, ti consigliamo di utilizzare Amazon EKS Pod Identity per migliorare l'isolamento delle credenziali, il privilegio minimo, la verificabilità, il funzionamento indipendente, la riusabilità e la scalabilità. Sia il tuo ruolo Pod che un ruolo IAM sono associati a un account di servizio Kubernetes per gestire le credenziali per le tue applicazioni. Per ulteriori informazioni su Amazon EKS Pod Identity, consulta [Amazon EKS Pod Identities](#) nella Amazon EKS User Guide. Una volta configurato, Amazon EKS imposta le variabili di ambiente `AWS_CONTAINER_CREDENTIALS_FULL_URI` e `AWS_CONTAINER_AUTHORIZATION_TOKEN_FILE` che gli SDK e gli strumenti utilizzano per ottenere le credenziali. Per informazioni sulla configurazione, consulta [Configurazione dell'agente Amazon EKS Pod Identity](#) nella Guida per l'utente di Amazon EKS o [Amazon EKS Pod Identity semplifica le autorizzazioni IAM per le applicazioni sui cluster Amazon EKS](#) sul sito Web del AWS blog.

Configura questa funzionalità utilizzando quanto segue:

#### **AWS\_CONTAINER\_CREDENTIALS\_FULL\_URI**- variabile di ambiente

Specifica l'endpoint URL HTTP completo per l'SDK da utilizzare quando si effettua una richiesta di credenziali. Ciò include sia lo schema che l'host.

Valore predefinito: Nessuno.

Valori validi: URI valido.

Nota: questa impostazione è un'alternativa `AWS_CONTAINER_CREDENTIALS_RELATIVE_URI` e verrà utilizzata solo se non `AWS_CONTAINER_CREDENTIALS_RELATIVE_URI` è impostata.

Esempio in Linux/macOS di impostazione delle variabili di ambiente tramite riga di comando:

```
export AWS_CONTAINER_CREDENTIALS_FULL_URI=http://localhost/get-credentials
```

oppure

```
export AWS_CONTAINER_CREDENTIALS_FULL_URI=http://localhost:8080/get-credentials
```

**AWS\_CONTAINER\_CREDENTIALS\_RELATIVE\_URI**- variabile di ambiente

Specifica l'endpoint URL HTTP relativo per l'SDK da utilizzare quando si effettua una richiesta di credenziali. Il valore viene aggiunto al nome host Amazon ECS predefinito di. 169.254.170.2

Valore predefinito: Nessuno.

Valori validi: URI relativo valido.

Esempio in Linux/macOS di impostazione delle variabili di ambiente tramite riga di comando:

```
export AWS_CONTAINER_CREDENTIALS_RELATIVE_URI=/get-credentials?a=1
```

**AWS\_CONTAINER\_AUTHORIZATION\_TOKEN**- variabile di ambiente

Specifica un token di autorizzazione in testo semplice. Se questa variabile è impostata, l'SDK imposterà l'intestazione Authorization sulla richiesta HTTP con il valore della variabile di ambiente.

Valore predefinito: Nessuno.

Valori validi: String.

Nota: questa impostazione è un'alternativa `AWS_CONTAINER_AUTHORIZATION_TOKEN_FILE` e verrà utilizzata solo se non `AWS_CONTAINER_AUTHORIZATION_TOKEN_FILE` è impostata.

Esempio in Linux/macOS di impostazione delle variabili di ambiente tramite riga di comando:

```
export AWS_CONTAINER_CREDENTIALS_FULL_URI=http://localhost/get-credential  
export AWS_CONTAINER_AUTHORIZATION_TOKEN=Basic abcd
```

**AWS\_CONTAINER\_AUTHORIZATION\_TOKEN\_FILE**- variabile di ambiente

Specifica il percorso assoluto di un file che contiene il token di autorizzazione in testo semplice.

Valore predefinito: Nessuno.

Valori validi: String.

Esempio in Linux/macOS di impostazione delle variabili di ambiente tramite riga di comando:

```
export AWS_CONTAINER_CREDENTIALS_FULL_URI=http://localhost/get-credential  
export AWS_CONTAINER_AUTHORIZATION_TOKEN_FILE=/path/to/token
```

## Compatibilità con AWS gli SDK

I seguenti SDK supportano le funzionalità e le impostazioni descritte in questo argomento. Vengono annotate eventuali eccezioni parziali. Tutte le impostazioni delle proprietà del sistema JVM sono supportate solo da AWS SDK for Java and the. SDK AWS for Kotlin

SDK	Sì o	Note o ulteriori informazioni
<a href="#">AWS CLI v2</a>	Sì	
<a href="#">SDK per C++</a>	Sì	
<a href="#">SDK per Go V2 (1.x)</a>	Sì	
<a href="#">SDK per Go 1.x (V1)</a>	Sì	
<a href="#">SDK per Java 2.x</a>	Sì	
<a href="#">SDK per Java 1.x</a>	Parziale	Amazon EKS Pod Identity e AWS_CONTAINER_AUTHORIZATION_TOKEN_FILE non sono supportati.
<a href="#">SDK per 3.x JavaScript</a>	Sì	
<a href="#">SDK per 2.x JavaScript</a>	Parziale	Amazon EKS Pod Identity e AWS_CONTAINER_AUTHORIZATION_TOKEN_FILE non sono supportati.
<a href="#">SDK per Kotlin</a>	Sì	
<a href="#">SDK per .NET 3.x</a>	Sì	
<a href="#">SDK per PHP 3.x</a>	Sì	
<a href="#">SDK per Python (Boto3)</a>	Sì	
<a href="#">SDK per Ruby 3.x</a>	Sì	
<a href="#">SDK per Rust</a>	Parziale	Amazon EKS Pod Identity e AWS_CONTAINER_AUTHORIZATION_TOKEN_FILE non sono supportati.
<a href="#">Strumenti per PowerShell</a>	Sì	



## Provider di credenziali IAM Identity Center

Questo meccanismo di autenticazione consente AWS IAM Identity Center di ottenere l'accesso Single Sign-On (SSO) al codice. Servizi AWS

### Note

Nella documentazione dell'API AWS SDK, il provider di credenziali IAM Identity Center è chiamato provider di credenziali SSO.

Dopo aver abilitato IAM Identity Center, definisci un profilo per le relative impostazioni nel file condiviso. `AWS config` Questo profilo viene utilizzato per connettersi al portale di accesso IAM Identity Center. Quando un utente si autentica con successo con IAM Identity Center, il portale restituisce credenziali a breve termine per il ruolo IAM associato a quell'utente. Per scoprire come l'SDK ottiene credenziali temporanee dalla configurazione e le utilizza per le richieste, consulta. Servizio AWS [Comprendi l'autenticazione IAM Identity Center](#)

Esistono due modi per configurare IAM Identity Center tramite il `config` file:

- Configurazione del provider di token SSO (consigliata): durate di sessione estese.
- Configurazione legacy non aggiornabile: utilizza una sessione fissa di otto ore.

In entrambe le configurazioni, è necessario accedere nuovamente alla scadenza della sessione.

Per impostare durate di sessione personalizzate, è necessario utilizzare la configurazione del provider di token SSO.

Le due guide seguenti contengono informazioni aggiuntive su IAM Identity Center:

- [AWS IAM Identity Center Guida per l'utente](#)
- [AWS IAM Identity Center Riferimento all'API del portale](#)

## Prerequisiti

Devi prima abilitare IAM Identity Center. Per i dettagli sull'attivazione dell'autenticazione IAM Identity Center, consulta la Guida [introduttiva](#) nella Guida AWS IAM Identity Center per l'utente.

In alternativa, segui le [Autenticazione IAM Identity Center](#) istruzioni contenute in questa guida. Queste istruzioni forniscono una guida completa, dall'attivazione di IAM Identity Center al completamento della necessaria configurazione dei config file condivisi che segue qui.

## Configurazione del provider di token SSO

### Note

Per utilizzare AWS CLI per creare questa configurazione per te, vedi [Configurare il tuo profilo con la aws configure sso procedura guidata](#) in AWS CLI

Quando utilizzi la configurazione del provider di token SSO, l'AWS SDK o lo strumento aggiorna automaticamente la sessione fino al periodo di sessione prolungato. Per ulteriori informazioni sulla durata della sessione e sulla durata massima, consulta [Configurare la durata della sessione del portale di AWS accesso e delle applicazioni integrate IAM Identity Center](#) nella Guida per l'AWS IAM Identity Center utente.

La `sso-session` sezione del config file viene utilizzata per raggruppare le variabili di configurazione per l'acquisizione dei token di accesso SSO, che possono quindi essere utilizzati per acquisire le credenziali. AWS Per ulteriori dettagli sulla formattazione delle sezioni all'interno di un file, vedere. config [Formato del file di configurazione](#)

Si definisce una `sso-session` sezione e la si associa a un profilo. `sso_regione` `sso_start_url` deve essere impostato all'interno della `sso-session` sezione. In genere, `sso_account_id` e `sso_role_name` deve essere impostato nella `profile` sezione in modo che l'SDK possa richiedere AWS le credenziali.

### Note

Per un'analisi approfondita su come gli SDK e gli strumenti utilizzano e aggiornano le credenziali utilizzando questa configurazione, consulta. [Comprendi l'autenticazione IAM Identity Center](#)

L'esempio seguente configura l'SDK per richiedere le credenziali IAM Identity Center. Supporta anche l'aggiornamento automatico dei token.

```
[profile dev]
```

```
sso_session = my-ss0
sso_account_id = 111122223333
sso_role_name = SampleRole

[sso-session my-ss0]
sso_region = us-east-1
sso_start_url = https://my-ss0-portal.awsapps.com/start
sso_registration_scopes = sso:account:access
```

È possibile riutilizzare le sso-session configurazioni su più profili.

```
[profile dev]
sso_session = my-ss0
sso_account_id = 111122223333
sso_role_name = SampleRole

[profile prod]
sso_session = my-ss0
sso_account_id = 111122223333
sso_role_name = SampleRole2

[sso-session my-ss0]
sso_region = us-east-1
sso_start_url = https://my-ss0-portal.awsapps.com/start
sso_registration_scopes = sso:account:access
```

sso\_account\_id e sso\_role\_name non sono necessari per tutti gli scenari di configurazione del token SSO. Se l'applicazione utilizza solo il supporto per Servizi AWS l'autenticazione al portatore, non sono necessarie AWS le credenziali tradizionali. L'autenticazione Bearer è uno schema di autenticazione HTTP che utilizza token di sicurezza chiamati bearer tokens. In questo scenario, sso\_account\_id e sso\_role\_name non sono obbligatori. Consulta la guida individuale per Servizio AWS determinare se supporta l'autorizzazione del token al portatore.

Gli ambiti di registrazione sono configurati come parte di un. sso-session L'ambito è un meccanismo OAuth 2.0 per limitare l'accesso di un'applicazione all'account di un utente. Un'applicazione può richiedere uno o più ambiti e il token di accesso rilasciato all'applicazione è limitato agli ambiti concessi. Questi ambiti definiscono le autorizzazioni richieste per l'autorizzazione per il client OIDC registrato e i token di accesso recuperati dal client. Per le opzioni di ambito di accesso supportate, consulta Ambiti di [accesso](#) nella Guida per l'utente.AWS IAM Identity Center L'esempio seguente fornisce l'accesso per sso\_registration\_scopes elencare account e ruoli.

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

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

## Configurazione legacy non aggiornabile

L'aggiornamento automatico dei token non è supportato utilizzando la configurazione legacy non aggiornabile. Ti consigliamo invece di utilizzare. [Configurazione del provider di token SSO](#)

Per utilizzare la configurazione precedente non aggiornabile, è necessario specificare le seguenti impostazioni all'interno del profilo:

- `sso_start_url`
- `sso_region`
- `sso_account_id`
- `sso_role_name`

Si specifica il portale utente per un profilo con le impostazioni `sso_start_url` and `sso_region`. Le autorizzazioni vengono specificate con le `sso_role_name` impostazioni `sso_account_id` e.

L'esempio seguente imposta i quattro valori obbligatori nel config file.

```
[profile my-sso-profile]  
sso_start_url = https://my-sso-portal.awsapps.com/start  
sso_region = us-west-2  
sso_account_id = 111122223333  
sso_role_name = SSOReadOnlyRole
```

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

## Impostazioni del provider di credenziali IAM Identity Center

Configura questa funzionalità utilizzando quanto segue:

**sso\_start\_url**- impostazione dei AWS **config** file condivisi

L'URL che rimanda al portale di accesso IAM Identity Center della tua organizzazione. Per ulteriori informazioni sul portale di accesso IAM Identity Center, consulta [Using the AWS access portal](#) nella AWS IAM Identity Center User Guide.

Per trovare questo valore, apri la [console IAM Identity Center](#), visualizza la dashboard e trova AWS l'URL del portale di accesso.

**sso\_region**- impostazione dei AWS **config** file condivisi

Il Regione AWS che contiene l'host del portale IAM Identity Center, ovvero la regione selezionata prima di abilitare IAM Identity Center. È indipendente dalla AWS regione predefinita e può essere diversa.

Per un elenco completo di Regioni AWS e dei relativi codici, consulta [Endpoint regionali](#) nei Riferimenti generali di Amazon Web Services. Per trovare questo valore, apri la [console IAM Identity Center](#), visualizza la dashboard e trova Region.

**sso\_account\_id**- impostazione dei AWS **config** file condivisi

L'ID numerico di Account AWS che è stato aggiunto tramite il AWS Organizations servizio da utilizzare per l'autenticazione.

Per visualizzare l'elenco degli account disponibili, vai alla [console IAM Identity Center](#) e apri la Account AWS pagina. Puoi anche visualizzare l'elenco degli account disponibili utilizzando il metodo [ListAccounts](#) API nel AWS IAM Identity Center Portal API Reference. Ad esempio, puoi chiamare il AWS CLI metodo [list-accounts](#).

**sso\_role\_name**- impostazione di file condivisi AWS **config**

Il nome di un set di autorizzazioni fornito come ruolo IAM che definisce le autorizzazioni risultanti dell'utente. Il ruolo deve esistere nel file Account AWS specificato da `sso_account_id` Usa il nome del ruolo, non il ruolo Amazon Resource Name (ARN).

I set di autorizzazioni sono associati a politiche IAM e politiche di autorizzazione personalizzate e definiscono il livello di accesso degli utenti ai dati loro assegnati. Account AWS

Per visualizzare l'elenco dei set di autorizzazioni disponibili per Account AWS, vai alla [console IAM Identity Center](#) e apri la Account AWS pagina. Scegli il nome corretto del set di autorizzazioni elencato nella Account AWS tabella. Puoi anche visualizzare l'elenco dei set di autorizzazioni

disponibili utilizzando il metodo [ListAccountRoles](#) API nel AWS IAM Identity Center Portal API Reference. Ad esempio, puoi chiamare il AWS CLI metodo [list-account-roles](#).

### **sso\_registration\_scopes**- impostazione di AWS **config** file condivisi

Un elenco delimitato da virgole di stringhe di ambito valide da autorizzare per. `sso-session` Gli ambiti autorizzano l'accesso agli endpoint autorizzati con token portatore di IAM Identity Center. È `sso:account:access` necessario concedere un ambito minimo di per ottenere un token di aggiornamento dal servizio IAM Identity Center. Per le stringhe di ambito di accesso supportate, consulta [Access scopes nella Guida](#) per l'AWS IAM Identity Center utente. Questa impostazione non si applica alla configurazione precedente non aggiornabile. I token emessi utilizzando la configurazione legacy sono limitati all'ambito implicito. `sso:account:access`

## Compatibilità con gli SDK AWS

I seguenti SDK supportano le funzionalità e le impostazioni descritte in questo argomento. Vengono annotate eventuali eccezioni parziali. Tutte le impostazioni delle proprietà del sistema JVM sono supportate solo da AWS SDK for Java and the. SDK AWS for Kotlin

SDK	Sì o	Note o ulteriori informazioni
<a href="#">AWS CLI v2</a>	Sì	
<a href="#">SDK per C++</a>	Sì	
<a href="#">SDK per Go V2 (1.x)</a>	Sì	
<a href="#">SDK per Go 1.x (V1)</a>	Sì	<a href="#">Per utilizzare le impostazioni dei config file condivisi, devi attivare il caricamento dal file di configurazione; vedi Sessioni.</a>
<a href="#">SDK per Java 2.x</a>	Sì	Valori di configurazione supportati anche nel <code>credentials</code> file.
<a href="#">SDK per Java 1.x</a>	No	
<a href="#">SDK per 3.x JavaScript</a>	Sì	
<a href="#">SDK per 2.x JavaScript</a>	Sì	

SDK	Sì o	Note o ulteriori informazioni
<a href="#">SDK per Kotlin</a>	Sì	
<a href="#">SDK per .NET 3.x</a>	Sì	
<a href="#">SDK per PHP 3.x</a>	Sì	
<a href="#">SDK per Python (Boto3)</a>	Sì	
<a href="#">SDK per Ruby 3.x</a>	Sì	
<a href="#">SDK per Rust</a>	Parziale	Solo configurazione precedente non aggiornabile.
<a href="#">Strumenti per PowerShell</a>	Sì	

## Provider di credenziali IMDS

Instance Metadata Service (IMDS) fornisce dati sull'istanza che puoi utilizzare per configurare o gestire l'istanza in esecuzione. Per ulteriori informazioni sui dati disponibili, consulta [Metadati e dati utente dell'istanza nella Guida per l'utente](#) di Amazon EC2 per le istanze Linux o Metadati e dati utente delle istanze nella Guida per [l'utente](#) di Amazon EC2 per istanze Windows. Amazon EC2 fornisce un endpoint locale disponibile per le istanze in grado di fornire vari bit di informazioni all'istanza. Se all'istanza è associato un ruolo, può fornire un set di credenziali valide per quel ruolo. Gli SDK possono utilizzare quell'endpoint per risolvere le credenziali come parte della catena di provider di credenziali [predefinita](#). Per impostazione predefinita, viene utilizzata la versione 2 di Instance Metadata Service (IMDSv2), una versione più sicura di IMDS che utilizza un token di sessione. Se ciò non riesce a causa di una condizione non riutilizzabile (codici di errore HTTP 403, 404, 405), IMDSv1 viene utilizzato come fallback.

Configura questa funzionalità utilizzando quanto segue:

### **AWS\_EC2\_METADATA\_DISABLED**- variabile di ambiente

Se tentare o meno di utilizzare Amazon EC2 Instance Metadata Service (IMDS) per ottenere le credenziali.

Valore predefinito: `false`.

Valori validi:


- **true**— Non utilizzare IMDS per ottenere credenziali.
- **false**— Usa IMDS per ottenere le credenziali.

**ec2\_metadata\_v1\_disabled**- impostazione di file condivisi AWS **config**,

**AWS\_EC2\_METADATA\_V1\_DISABLED**- variabile d'ambiente, **aws.disableEc2MetadataV1**-

Proprietà del sistema JVM: solo Java/Kotlin

Se utilizzare o meno Instance Metadata Service Version 1 (IMDSv1) come fallback in caso di errore di IMDSv2.

 Note

I nuovi SDK non supportano IMDSv1 e, pertanto, non supportano questa impostazione. Per i dettagli, consulta la tabella. [Compatibilità con gli SDK AWS](#)

Valore predefinito: `false`.

Valori validi:

- **true**— Non utilizzare IMDSv1 come fallback.
- **false**— Usa IMDSv1 come fallback.

**ec2\_metadata\_service\_endpoint**- impostazione condivisa dei file AWS

**config**, **AWS\_EC2\_METADATA\_SERVICE\_ENDPOINT**- variabile d'ambiente,

**aws.ec2MetadataServiceEndpoint**- Proprietà del sistema JVM: solo Java/Kotlin

L'endpoint di IMDS.

Valore predefinito: se è `ec2_metadata_service_endpoint_mode` uguale `IPv4`, l'endpoint predefinito è. `http://169.254.169.254` Se è `ec2_metadata_service_endpoint_mode` uguale, l'endpoint predefinito è. IPv6 `http://[fd00:ec2::254]`

Valori validi: URI valido.

**ec2\_metadata\_service\_endpoint\_mode**- impostazione di AWS **config** file

condivisi, **AWS\_EC2\_METADATA\_SERVICE\_ENDPOINT\_MODE**- variabile d'ambiente,

**aws.ec2MetadataServiceEndpointMode**- Proprietà del sistema JVM: solo Java/Kotlin

La modalità endpoint di IMDS.

Valore predefinito: `IPv4`



Valori validi: IPv4, IPv6.

### Note

Il provider di credenziali IMDS fa parte di [Catena di fornitori di credenziali](#). Tuttavia, il fornitore di credenziali IMDS viene controllato solo dopo diversi altri provider di questa serie. Pertanto, se si desidera che il programma utilizzi le credenziali di questo provider, è necessario rimuovere altri provider di credenziali validi dalla configurazione o utilizzare un profilo diverso. In alternativa, anziché affidarsi alla catena di fornitori di credenziali per scoprire automaticamente quale provider restituisce credenziali valide, specifica l'uso del provider di credenziali IMDS nel codice. È possibile specificare le fonti delle credenziali direttamente quando si creano client di servizio.

## Sicurezza per le credenziali IMDS

Per impostazione predefinita, quando l'AWS SDK non è configurato con credenziali valide, l'SDK tenterà di utilizzare Amazon EC2 Instance Metadata Service (IMDS) per recuperare le credenziali per un ruolo. AWS Questo comportamento può essere disabilitato impostando la variabile di ambiente su `AWS_EC2_METADATA_DISABLED true`. Ciò impedisce attività di rete non necessarie e migliora la sicurezza su reti non attendibili in cui l'Amazon EC2 Instance Metadata Service può essere impersonato.

### Note

AWS I client SDK configurati con credenziali valide non utilizzeranno mai IMDS per recuperare le credenziali, indipendentemente da nessuna di queste impostazioni.

## Disabilitazione dell'uso delle credenziali IMDS di Amazon EC2

Il modo in cui imposti questa variabile di ambiente dipende dal sistema operativo in uso e dal fatto che desideri o meno che la modifica sia persistente.

### Linux e macOS

I clienti che utilizzano Linux o macOS possono impostare questa variabile di ambiente con il seguente comando:

```
$ export AWS_EC2_METADATA_DISABLED=true
```

Se desideri che questa impostazione sia persistente tra più sessioni di shell e riavvii del sistema, puoi aggiungere il comando precedente al file del profilo della shell, ad esempio `.bash_profile`, `.zsh_profile`, o `.profile`

## Windows

I clienti che utilizzano Windows possono impostare questa variabile di ambiente con il seguente comando:

```
$ set AWS_EC2_METADATA_DISABLED=true
```

Se desideri che questa impostazione sia persistente tra più sessioni di shell e riavvii del sistema, puoi utilizzare invece il seguente comando:

```
$ setx AWS_EC2_METADATA_DISABLED=true
```

### Note

Il `setx` comando non applica il valore alla sessione di shell corrente, quindi sarà necessario ricaricare o riaprire la shell affinché la modifica abbia effetto.

## Compatibilità con gli SDK AWS

I seguenti SDK supportano le funzionalità e le impostazioni descritte in questo argomento. Vengono annotate eventuali eccezioni parziali. Tutte le impostazioni delle proprietà del sistema JVM sono supportate solo da AWS SDK for Java and the. SDK AWS for Kotlin

SDK	Sì o	Note o ulteriori informazioni
<a href="#">AWS CLI v2</a>	Sì	
<a href="#">SDK per C++</a>	Sì	

SDK	Sì	Note o ulteriori informazioni
<a href="#">SDK per Go V2 (1.x)</a>	Sì	
<a href="#">SDK per Go 1.x (V1)</a>	Sì	<a href="#">Per utilizzare le impostazioni dei config file condivisi, devi attivare il caricamento dal file di configurazione; vedi Sessioni.</a>
<a href="#">SDK per Java 2.x</a>	Sì	
<a href="#">SDK per Java 1.x</a>	Sì	
<a href="#">SDK per 3.x JavaScript</a>	Sì	
<a href="#">SDK per 2.x JavaScript</a>	Sì	
<a href="#">SDK per Kotlin</a>	Sì	Non utilizza il fallback IMDSv1.
<a href="#">SDK per .NET 3.x</a>	Sì	
<a href="#">SDK per PHP 3.x</a>	Sì	
<a href="#">SDK per Python (Boto3)</a>	Sì	
<a href="#">SDK per Ruby 3.x</a>	Sì	
<a href="#">SDK per Rust</a>	Sì	Non utilizza il fallback IMDSv1.
<a href="#">Strumenti per PowerShell</a>	Sì	È possibile disabilitare il fallback IMDSv1 in modo esplicito nel codice utilizzando. <code>[Amazon.Util.EC2InstanceMetadata]::EC2MetadataV1Disabled = \$true</code>

## Provider di credenziali di processo

Gli SDK forniscono un modo per estendere la catena di fornitori di credenziali per casi d'uso personalizzati.

IAM Roles Anywhere offre un modo per ottenere credenziali temporanee per un carico di lavoro o un processo eseguito all'esterno di AWS. Per configurarlo `credential_process` per questo utilizzo, consulta [IAM Roles Anywhere](#)

**⚠ Warning**

Di seguito viene descritto un metodo di acquisizione delle credenziali da un processo esterno. Questo può essere potenzialmente pericoloso, quindi procedi con cautela. Se possibile, dovrebbero essere preferiti altri fornitori di credenziali. Se si utilizza questa opzione, è necessario assicurarsi che il `config` file sia il più protetto possibile utilizzando le migliori pratiche di sicurezza per il sistema operativo in uso. Verifica che lo strumento per le credenziali personalizzato non scriva informazioni segrete `stderr`, poiché gli SDK AWS CLI possono acquisire e registrare tali informazioni, esponendole potenzialmente a utenti non autorizzati.

Configura questa funzionalità utilizzando quanto segue:

**credential\_process**- impostazione dei AWS **config** file condivisi

Specifica un comando esterno che l'SDK o lo strumento esegue per conto dell'utente per generare o recuperare le credenziali di autenticazione da utilizzare. L'impostazione specifica il nome di un programma/comando che verrà richiamato dall'SDK. Quando l'SDK richiama il processo, attende che il processo scriva i dati JSON. `stdout` Il provider personalizzato deve restituire le informazioni in un formato specifico. Tali informazioni contengono le credenziali che l'SDK o lo strumento possono utilizzare per autenticare l'utente.

**📘 Note**

Il provider delle credenziali di processo fa parte di [Catena di fornitori di credenziali](#). Tuttavia, il fornitore delle credenziali di processo viene controllato solo dopo diversi altri provider di questa serie. Pertanto, se si desidera che il programma utilizzi le credenziali di questo provider, è necessario rimuovere altri provider di credenziali validi dalla configurazione o utilizzare un profilo diverso. In alternativa, anziché affidarsi alla catena di fornitori di credenziali per scoprire automaticamente quale provider restituisce credenziali valide, specificate l'uso del provider di credenziali di processo nel codice. È possibile specificare le fonti delle credenziali direttamente quando si creano client di servizio.

## Specificare il percorso del programma di credenziali

Il valore dell'impostazione è una stringa che contiene il percorso di un programma che l'SDK o lo strumento di sviluppo esegue per conto dell'utente:

- Il percorso e il nome del file possono essere composti solo dai seguenti caratteri: A-Z, a-z, 0-9, trattino (-), trattino basso (\_), punto (.), barra (/), barra rovesciata (\) e spazio.
- Se il percorso o il nome del file contiene uno spazio, circondare il percorso completo e il nome del file con virgolette doppie (" ").
- Se un nome di parametro o un valore di parametro contiene uno spazio, circondare tale elemento con virgolette doppie (" "). È possibile racchiudere solo il nome o il valore, non la coppia.
- Non includere alcuna variabile di ambiente nelle stringhe. Ad esempio, non includere \$HOME o %USERPROFILE%.
- Non specificare la cartella home come ~. \* È necessario specificare il percorso completo o il nome del file di base. Se è presente un nome di file di base, il sistema tenta di trovare il programma all'interno delle cartelle specificate dalla variabile di PATH ambiente.

L'esempio seguente mostra l'impostazione di `credential_process` nel file condiviso `config` su Linux/macOS.

```
credential_process = "/path/to/credentials.sh" parameterWithoutSpaces "parameter with spaces"
```

L'esempio seguente mostra l'impostazione di `credential_process` nel file condiviso su Windows.

`config`

```
credential_process = "C:\Path\To\credentials.cmd" parameterWithoutSpaces "parameter with spaces"
```

## Output valido dal programma di credenziali

L'SDK esegue il comando come specificato nel profilo e quindi legge i dati dal flusso di output standard. Il comando specificato, che si tratti di uno script o di un programma binario, deve generare un output JSON STDOUT che corrisponda alla sintassi seguente.

```
{  
  "Version": 1,  
}
```

```

"AccessKeyId": "an AWS access key",
"SecretAccessKey": "your AWS secret access key",
"SessionToken": "the AWS session token for temporary credentials",
"Expiration": "RFC3339 timestamp for when the credentials expire"
}

```

### Note

Al momento della stesura del presente documento, la chiave `Version` deve essere configurata su 1. Questo valore potrebbe incrementare nel tempo, man mano che la struttura evolve.

La `Expiration` chiave è un timestamp in formato RFC3339. Se la `Expiration` chiave non è presente nell'output dello strumento, l'SDK presuppone che le credenziali siano credenziali a lungo termine che non si aggiornano. Altrimenti, le credenziali sono considerate credenziali temporanee e vengono aggiornate automaticamente eseguendo nuovamente il comando prima della scadenza delle credenziali. `credential_process`

### Note

L'SDK non memorizza nella cache le credenziali dei processi esterni nello stesso modo in cui utilizza le credenziali di ruolo. Se il caching è necessario, dovrai implementarlo nel processo esterno.

Il processo esterno può restituire un codice diverso da zero per indicare che si è verificato un errore durante il recupero delle credenziali.

## AWS Compatibilità con gli SDK

I seguenti SDK supportano le funzionalità e le impostazioni descritte in questo argomento. Vengono annotate eventuali eccezioni parziali. Tutte le impostazioni delle proprietà del sistema JVM sono supportate solo da AWS SDK for Java and the. SDK AWS for Kotlin

SDK	Sì Note o ulteriori informazioni o
<a href="#">AWS CLI v2</a>	Sì

SDK	Sì	Note o ulteriori informazioni
<a href="#">SDK per C++</a>	Sì	
<a href="#">SDK per Go V2 (1.x)</a>	Sì	
<a href="#">SDK per Go 1.x (V1)</a>	Sì	<a href="#">Per utilizzare le impostazioni dei config file condivisi, devi attivare il caricamento dal file di configurazione; vedi Sessioni.</a>
<a href="#">SDK per Java 2.x</a>	Sì	
<a href="#">SDK per Java 1.x</a>	Sì	
<a href="#">SDK per 3.x JavaScript</a>	Sì	
<a href="#">SDK per 2.x JavaScript</a>	Sì	
<a href="#">SDK per Kotlin</a>	Sì	
<a href="#">SDK per .NET 3.x</a>	Sì	
<a href="#">SDK per PHP 3.x</a>	Sì	
<a href="#">SDK per Python (Boto3)</a>	Sì	
<a href="#">SDK per Ruby 3.x</a>	Sì	
<a href="#">SDK per Rust</a>	Sì	
<a href="#">Strumenti per PowerShell</a>	Sì	

## Funzionalità standardizzate

Molte funzionalità sono state standardizzate in base a impostazioni predefinite coerenti e per funzionare allo stesso modo su molti SDK. Questa coerenza aumenta la produttività e la chiarezza durante la codifica su più SDK. Tutte le impostazioni possono essere sovrascritte nel codice, consulta la tua API SDK specifica per i dettagli.

**⚠ Important**

Non tutti gli SDK supportano tutte le funzionalità o anche tutti gli aspetti all'interno di una funzionalità.

## Argomenti

- [Metadati delle istanze Amazon EC2.](#)
- [Punti di accesso Amazon S3](#)
- [Punti di accesso multi-Regione di Amazon S3](#)
- [Regione AWS](#)
- [AWS STS Endpoint regionalizzati](#)
- [Endpoint dual-stack e FIPS](#)
- [Rilevamento di endpoint](#)
- [Impostazioni generali di configurazione](#)
- [Cliente IMDS](#)
- [Comportamento di ripetizione](#)
- [Richiesta di compressione](#)
- [Endpoint specifici del servizio](#)
- [Impostazioni predefinite di configurazione intelligente](#)

## Metadati delle istanze Amazon EC2.

Amazon EC2 fornisce un servizio su istanze chiamato Instance Metadata Service (IMDS). Per ulteriori informazioni su questo servizio, consulta [i metadati delle istanze e i dati utente](#) nella Guida per l'utente di Amazon EC2 per le istanze Linux o [i metadati delle istanze e i dati utente](#) nella Amazon EC2 User Guide for Windows Instances. Quando si tenta di recuperare le credenziali su un'istanza Amazon EC2 configurata con un ruolo IAM, la connessione al servizio di metadati dell'istanza è regolabile.

Configura questa funzionalità utilizzando quanto segue:



**metadata\_service\_num\_attempts**- impostazione dei file **AWS config** condivisi,  
**AWS\_METADATA\_SERVICE\_NUM\_ATTEMPTS**- variabile d'ambiente

Questa impostazione specifica il numero totale di tentativi da effettuare prima di rinunciare al tentativo di recuperare dati dal servizio di metadati dell'istanza.

Valore predefinito: 1

Valori validi: numero maggiore o uguale a 1.

**metadata\_service\_timeout**- impostazione condivisa **AWS config** dei file,  
**AWS\_METADATA\_SERVICE\_TIMEOUT**- variabile d'ambiente

Specifica il numero di secondi prima del timeout quando si tenta di recuperare i dati dal servizio di metadati dell'istanza.

Valore predefinito: 1

Valori validi: numero maggiore o uguale a 1.

Esempio di impostazione di questi valori nel **config** file:

```
[default]
metadata_service_num_attempts=10
metadata_service_timeout=10
```

Esempio in Linux/macOS di impostazione delle variabili di ambiente tramite riga di comando:

```
export AWS_METADATA_SERVICE_NUM_ATTEMPTS=10
export AWS_METADATA_SERVICE_TIMEOUT=10
```

Esempio in Windows di impostazione delle variabili di ambiente tramite riga di comando:

```
setx AWS_METADATA_SERVICE_NUM_ATTEMPTS 10
setx AWS_METADATA_SERVICE_TIMEOUT 10
```

## Compatibilità con gli AWS SDK

I seguenti SDK supportano le funzionalità e le impostazioni descritte in questo argomento. Vengono annotate eventuali eccezioni parziali. Tutte le impostazioni delle proprietà del sistema JVM sono supportate solo da AWS SDK for Java and the. SDK AWS for Kotlin

SDK	Sì	Note o ulteriori informazioni
<a href="#">AWS CLI v2</a>	Sì	
<a href="#">SDK per C++</a>	No	
<a href="#">SDK per Go V2 (1.x)</a>	No	
<a href="#">SDK per Go 1.x (V1)</a>	No	
<a href="#">SDK per Java 2.x</a>	No	
<a href="#">SDK per Java 1.x</a>	Parziale	metadata_service_num_attempts non supportato.
<a href="#">SDK per 3.x JavaScript</a>	No	
<a href="#">SDK per 2.x JavaScript</a>	No	
<a href="#">SDK per Kotlin</a>	No	
<a href="#">SDK per .NET 3.x</a>	No	
<a href="#">SDK per PHP 3.x</a>	Sì	
<a href="#">SDK per Python (Boto3)</a>	Sì	
<a href="#">SDK per Ruby 3.x</a>	No	
<a href="#">SDK per Rust</a>	No	
<a href="#">Strumenti per PowerShell</a>	No	

## Punti di accesso Amazon S3

Il servizio Amazon S3 fornisce punti di accesso come modo alternativo per interagire con i bucket Amazon S3. Gli access point hanno politiche e configurazioni uniche che possono essere applicate a loro anziché direttamente al bucket. Con AWS gli SDK, puoi utilizzare il punto di accesso Amazon Resource Names (ARN) nel campo del bucket per le operazioni API invece di specificare il nome del bucket in modo esplicito. Vengono utilizzati per operazioni specifiche come l'utilizzo di un punto di

accesso ARN [GetObject](#) per recuperare un oggetto da un bucket o l'utilizzo di un punto di accesso ARN [PutObject](#) per aggiungere un oggetto a un bucket.

Per ulteriori informazioni sui punti di accesso e gli ARN di Amazon S3, consulta [Using access point](#) nella Amazon S3 User Guide.

Configura questa funzionalità utilizzando quanto segue:

**s3\_use\_arn\_region**- impostazione dei AWS **config** file condivisi, **AWS\_S3\_USE\_ARN\_REGION**- variabile d'ambiente, **aws.s3UseArnRegion**- Proprietà del sistema JVM: solo Java/Kotlin, Per configurare il valore direttamente nel codice, consulta direttamente il tuo SDK specifico.

Questa impostazione controlla se l'SDK utilizza l' Regione AWS ARN del punto di accesso per costruire l'endpoint regionale per la richiesta. L'SDK verifica che l'ARN Regione AWS sia servito dalla stessa AWS partizione configurata dal client Regione AWS per evitare chiamate tra partizioni che molto probabilmente falliranno. Se definita in modo multiplo, l'impostazione configurata dal codice ha la precedenza, seguita dall'impostazione della variabile di ambiente.

Valore predefinito: `false`

Valori validi:

- **true**— L'SDK utilizza gli ARN Regione AWS durante la costruzione dell'endpoint anziché quelli configurati dal client. Regione AWS Eccezione: se la configurazione del client Regione AWS è un FIPS Regione AWS, deve corrispondere a quella dell'ARN. Regione AWS In caso contrario verrà restituito un errore.
- **false**— L'SDK utilizza la configurazione del client Regione AWS durante la costruzione dell'endpoint.

## Compatibilità con gli SDK AWS

I seguenti SDK supportano le funzionalità e le impostazioni descritte in questo argomento. Vengono annotate eventuali eccezioni parziali. Tutte le impostazioni delle proprietà del sistema JVM sono supportate solo da AWS SDK for Java and the. SDK AWS for Kotlin

SDK	Sì	Note o ulteriori informazioni
<a href="#">AWS CLI v2</a>	Sì	

SDK	Sì	Note o ulteriori informazioni
<a href="#">SDK per C++</a>	Sì	
<a href="#">SDK per Go V2 (1.x)</a>	Sì	
<a href="#">SDK per Go 1.x (V1)</a>	Sì	<a href="#">Per utilizzare le impostazioni dei config file condivisi, devi attivare il caricamento dal file di configurazione; vedi Sessioni.</a>
<a href="#">SDK per Java 2.x</a>	Sì	
<a href="#">SDK per Java 1.x</a>	Sì	
<a href="#">SDK per 3.x JavaScript</a>	Sì	
<a href="#">SDK per 2.x JavaScript</a>	Sì	
<a href="#">SDK per Kotlin</a>	Sì	
<a href="#">SDK per .NET 3.x</a>	Sì	Non segue la precedenza standard; il valore del config file condiviso ha la precedenza sulla variabile di ambiente.
<a href="#">SDK per PHP 3.x</a>	Sì	
<a href="#">SDK per Python (Boto3)</a>	Sì	
<a href="#">SDK per Ruby 3.x</a>	Sì	
<a href="#">SDK per Rust</a>	No	
<a href="#">Strumenti per PowerShell</a>	Sì	Non segue la precedenza standard; il valore config del file condiviso ha la precedenza sulla variabile di ambiente.

## Punti di accesso multi-Regione di Amazon S3

Gli access point multiregionali di Amazon S3 forniscono un endpoint globale che le applicazioni possono utilizzare per soddisfare le richieste provenienti da bucket Amazon S3 distribuiti in più aree. Regioni AWS È possibile utilizzare punti di accesso multiregionali per creare applicazioni

multiregionali con la stessa architettura utilizzata in una singola regione e quindi eseguire tali applicazioni in qualsiasi parte del mondo.

Per ulteriori informazioni sui punti di accesso multiregionali, consulta Punti di [accesso multiregionali in Amazon S3 nella Guida](#) per l'utente di Amazon S3.

Per ulteriori informazioni sui nomi di risorse Amazon Resource Names (ARN) per punti di accesso multiregionali, consulta [Effettuare richieste utilizzando un punto di accesso multiregionale nella Guida](#) per l'utente di Amazon S3.

Per ulteriori informazioni sulla creazione di punti di accesso multiregionali, consulta [Managing Multi-Region Access Points](#) nella Amazon S3 User Guide.

L'algoritmo SigV4A è l'implementazione di firma utilizzata per firmare le richieste regionali globali. Questo algoritmo è ottenuto dall'SDK tramite una dipendenza da [AWSLibrerie Common Runtime \(CRT\)](#)

Configura questa funzionalità utilizzando quanto segue:

**s3\_disable\_multiregion\_access\_points**- impostazione dei AWS **config** file condivisi, **AWS\_S3\_DISABLE\_MULTIREGION\_ACCESS\_POINTS**- variabile d'ambiente, **aws.s3DisableMultiRegionAccessPoints**- Proprietà del sistema JVM: solo Java/Kotlin, Per configurare il valore direttamente nel codice, consulta direttamente il tuo SDK specifico.

Questa impostazione controlla se l'SDK tenta potenzialmente di effettuare richieste interregionali. Se definita in modo multiplo, l'impostazione configurata dal codice ha la precedenza, seguita dall'impostazione della variabile di ambiente.

Valore predefinito: `false`

Valori validi:

- **true**— Interrompe l'uso delle richieste interregionali.
- **false**— Abilita le richieste interregionali utilizzando punti di accesso multiregionali.

## Compatibilità con gli SDK AWS

I seguenti SDK supportano le funzionalità e le impostazioni descritte in questo argomento. Vengono annotate eventuali eccezioni parziali. Tutte le impostazioni delle proprietà del sistema JVM sono supportate solo da AWS SDK for Java and the. SDK AWS for Kotlin

SDK	Sì o	Note o ulteriori informazioni
<a href="#">AWS CLI v2</a>	Sì	
<a href="#">SDK per C++</a>	Sì	
<a href="#">SDK per Go V2 (1.x)</a>	Sì	
<a href="#">SDK per Go 1.x (V1)</a>	No	
<a href="#">SDK per Java 2.x</a>	Sì	
<a href="#">SDK per Java 1.x</a>	No	
<a href="#">SDK per 3.x JavaScript</a>	Sì	
<a href="#">SDK per 2.x JavaScript</a>	No	
<a href="#">SDK per Kotlin</a>	Sì	
<a href="#">SDK per .NET 3.x</a>	Sì	
<a href="#">SDK per PHP 3.x</a>	Sì	
<a href="#">SDK per Python (Boto3)</a>	Sì	
<a href="#">SDK per Ruby 3.x</a>	Sì	
<a href="#">SDK per Rust</a>	Sì	
<a href="#">Strumenti per PowerShell</a>	Sì	

## Regione AWS

Regioni AWS sono un concetto importante da comprendere quando si lavora Servizi AWS.

Con Regioni AWS, puoi accedere a chi Servizi AWS risiede fisicamente in un'area geografica specifica. Ciò può essere utile per mantenere attivi i dati e le applicazioni in prossimità del luogo in cui voi e i vostri utenti potrete accedervi. Le regioni forniscono la tolleranza ai guasti, la stabilità e

la resilienza e possono anche ridurre la latenza. Con Regions, puoi creare risorse ridondanti che rimangono disponibili e non sono interessate da un'interruzione regionale.

La maggior parte delle Servizio AWS richieste è associata a una particolare area geografica. Le risorse create in una regione non esistono in nessun'altra regione a meno che non si utilizzi esplicitamente una funzionalità di replica offerta da un. Servizio AWS Ad esempio, Amazon S3 e Amazon EC2 supportano la replica tra regioni. Alcuni servizi, come IAM, non dispongono di risorse regionali.

Riferimenti generali di AWS Contiene informazioni su quanto segue:

- Per comprendere la relazione tra le regioni e gli endpoint e per visualizzare un elenco degli endpoint regionali esistenti, consulta [AWS Service Endpoint](#).
- Per visualizzare l'elenco corrente di tutte le regioni e gli endpoint supportati per ciascuna Servizio AWS, consulta Endpoint e quote [del servizio](#).

## Creazione di client di servizio

Per accedere a livello di codice Servizi AWS, gli SDK utilizzano una classe/oggetto client per ciascuno. Servizio AWS Se l'applicazione deve accedere ad Amazon EC2, ad esempio, l'applicazione creerà un oggetto client Amazon EC2 per interfacciarsi con quel servizio.

Se non viene specificata in modo esplicito alcuna regione per il client, per impostazione predefinita il client utilizza la regione impostata tramite l'impostazione seguente. `region` Tuttavia, la regione attiva per un client può essere impostata in modo esplicito per ogni singolo oggetto client. L'impostazione della Regione in questo modo ha la precedenza su qualsiasi impostazione globale per quel particolare client di servizio. La regione alternativa viene specificata durante la creazione di un'istanza di quel client, specifica per il tuo SDK (consulta la guida SDK specifica o la base di codice dell'SDK).

Configura questa funzionalità utilizzando quanto segue:

**region**- impostazione dei AWS **config** file condivisi, **AWS\_REGION**- variabile d'ambiente, **aws.region**- Proprietà del sistema JVM: solo Java/Kotlin

Specifica l'impostazione predefinita da utilizzare per le richieste. Regione AWS AWS Questa regione viene utilizzata per le richieste di servizio SDK a cui non viene fornita una regione specifica da utilizzare.

Valore predefinito: Nessuno. È necessario specificare questo valore in modo esplicito.

### Valori validi:

- Tutti i codici regionali disponibili per il servizio scelto, elencati negli [endpoint del AWS servizio](#) nel Riferimento AWS generale. Ad esempio, il valore `us-east-1` imposta l'endpoint sugli Regione AWS Stati Uniti orientali (Virginia settentrionale).
- `aws-global` specifica l'endpoint globale per i servizi che supportano un endpoint globale separato oltre agli endpoint regionali, come AWS Security Token Service (AWS STS) e Amazon Simple Storage Service (Amazon S3).

Esempio di impostazione di questo valore nel file: `config`

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

Esempio in Linux/macOS di impostazione delle variabili di ambiente tramite riga di comando:

```
export AWS_REGION=us-west-2
```

Esempio in Windows di impostazione delle variabili di ambiente tramite riga di comando:

```
setx AWS_REGION us-west-2
```

La maggior parte degli SDK dispone di un oggetto di «configurazione» che è disponibile per impostare la regione predefinita all'interno del codice dell'applicazione. Per i dettagli, consulta la tua guida per sviluppatori AWS SDK specifica.

## Compatibilità con AWS gli SDK

I seguenti SDK supportano le funzionalità e le impostazioni descritte in questo argomento. Vengono annotate eventuali eccezioni parziali. Tutte le impostazioni delle proprietà del sistema JVM sono supportate solo da AWS SDK for Java and the. SDK AWS for Kotlin

SDK	Sì o No	Note o ulteriori informazioni
<a href="#">AWS CLI v2</a>	Sì	AWS CLI v2 utilizza qualsiasi valore <code>AWS_REGION</code> prima di qualsiasi valore in <code>AWS_DEFAULT_REGION</code> (entrambe le variabili vengono controllate).



SDK	Sì	Note o ulteriori informazioni
<a href="#">AWS CLI v1</a>	Sì	AWS CLI v1 utilizza una variabile di ambiente denominata a questo scopo. <code>AWS_DEFAULT_REGION</code>
<a href="#">SDK per C++</a>	Sì	
<a href="#">SDK per Go V2 (1.x)</a>	Sì	
<a href="#">SDK per Go 1.x (V1)</a>	Sì	<a href="#">Per utilizzare le impostazioni dei config file condivisi, devi attivare il caricamento dal file di configurazione; vedi Sessioni.</a>
<a href="#">SDK per Java 2.x</a>	Sì	
<a href="#">SDK per Java 1.x</a>	Sì	
<a href="#">SDK per 3.x JavaScript</a>	Sì	
<a href="#">SDK per 2.x JavaScript</a>	Sì	
<a href="#">SDK per Kotlin</a>	Sì	
<a href="#">SDK per .NET 3.x</a>	Sì	
<a href="#">SDK per PHP 3.x</a>	Sì	
<a href="#">SDK per Python (Boto3)</a>	Sì	Questo SDK utilizza una variabile di ambiente denominata a questo scopo. <code>AWS_DEFAULT_REGION</code>
<a href="#">SDK per Ruby 3.x</a>	Sì	
<a href="#">SDK per Rust</a>	Sì	
<a href="#">Strumenti per PowerShell</a>	Sì	

## AWS STS Endpoint regionalizzati

Per impostazione predefinita, AWS Security Token Service (AWS STS) è disponibile come servizio globale e tutte le AWS STS richieste vanno a un singolo endpoint all'indirizzo. `https://`

`sts.amazonaws.com` Le richieste globali si riferiscono alla regione Stati Uniti orientali (Virginia settentrionale). AWS consiglia di utilizzare gli AWS STS endpoint regionali anziché l'endpoint globale. Per ulteriori informazioni sugli AWS STS endpoint, consulta [Endpoints nell'API Reference](#). AWS Security Token Service


Configura questa funzionalità utilizzando quanto segue:

**sts\_regional\_endpoints**- impostazione dei AWS **config** file condivisi,

**AWS\_STS\_REGIONAL\_ENDPOINTS**- variabile d'ambiente

Questa impostazione specifica in che modo l'SDK o lo strumento determina l' Servizio AWS endpoint che utilizza per comunicare con (). AWS Security Token Service AWS STS

Valore predefinito: `legacy`

 Note

Tutte le nuove versioni principali dell'SDK rilasciate dopo luglio 2022 verranno utilizzate per impostazione predefinita. `regional` Le nuove versioni principali dell'SDK potrebbero rimuovere questa impostazione e questo comportamento d'uso. `regional` Per ridurre l'impatto futuro di questa modifica, ti consigliamo di iniziare a `regional` utilizzarla nella tua applicazione quando possibile.

Valori validi: (Valore consigliato: `regional`)

- **legacy**— Utilizza l' AWS STS endpoint globale `sts.amazonaws.com`, per le seguenti AWS regioni: `ap-northeast-1`, `ap-south-1`, `ap-southeast-1`, `ap-southeast-2`, `aws-global`, `ca-central-1`, `eu-central-1`, `eu-north-1`, `eu-west-1`, `eu-west-2`, `eu-west-3`, `sa-east-1`, `us-east-1`, `us-east-2`, `us-west-1`, `us-west-2`. Tutte le altre regioni utilizzano automaticamente il rispettivo endpoint regionale.
- **regional**— L'SDK o lo strumento utilizza sempre l' AWS STS endpoint per la regione attualmente configurata. Ad esempio, se il client è configurato per l'uso `us-west-2`, tutte le chiamate AWS STS vengono effettuate all'endpoint regionale `sts.us-west-2.amazonaws.com`, anziché all'endpoint globale `sts.amazonaws.com`. Per inviare una richiesta all'endpoint globale mentre questa impostazione è abilitata, è possibile impostare l'area geografica su `aws-global`.

Esempio di impostazione di questi valori nel `config` file:

```
[default]
sts_regional_endpoints = regional
```

Esempio in Linux/macOS di impostazione delle variabili di ambiente tramite riga di comando:

```
export AWS_STS_REGIONAL_ENDPOINTS=regional
```

Esempio in Windows di impostazione delle variabili di ambiente tramite riga di comando:

```
setx AWS_STS_REGIONAL_ENDPOINTS regional
```

## Compatibilità con gli AWS SDK

I seguenti SDK supportano le funzionalità e le impostazioni descritte in questo argomento. Vengono annotate eventuali eccezioni parziali. Tutte le impostazioni delle proprietà del sistema JVM sono supportate solo da AWS SDK for Java and the. SDK AWS for Kotlin

SDK	Sì o	Note o ulteriori informazioni
<a href="#">AWS CLI v2</a>	Parziale	Il valore predefinito è <code>regional</code> .
<a href="#">SDK per C++</a>	Parziale	Variabile di ambiente e impostazione config dei file non supportate. L'SDK funziona con l' <code>regional</code> impostazione.
<a href="#">SDK per Go V2 (1.x)</a>	Sì	
<a href="#">SDK per Go 1.x (V1)</a>	Sì	<a href="#">Per utilizzare le impostazioni dei config file condivisi, devi attivare il caricamento dal file di configurazione; vedi Sessioni.</a>
<a href="#">SDK per Java 2.x</a>	Sì	
<a href="#">SDK per Java 1.x</a>	Sì	
<a href="#">SDK per 3.x JavaScript</a>	Sì	
<a href="#">SDK per 2.x JavaScript</a>	Sì	
<a href="#">SDK per Kotlin</a>	No	

SDK	Sì o	Note o ulteriori informazioni
<a href="#">SDK per .NET 3.x</a>	Sì	
<a href="#">SDK per PHP 3.x</a>	Sì	
<a href="#">SDK per Python (Boto3)</a>	Sì	
<a href="#">SDK per Ruby 3.x</a>	Sì	
<a href="#">SDK per Rust</a>	Sì	
<a href="#">Strumenti per PowerShell</a>	Sì	

## Endpoint dual-stack e FIPS

Configura questa funzionalità utilizzando quanto segue:

**use\_dualstack\_endpoint**- impostazione dei AWS **config** file condivisi,

**AWS\_USE\_DUALSTACK\_ENDPOINT**- variabile d'ambiente, **aws.useDualstackEndpoint**- Proprietà del sistema JVM: solo Java/Kotlin

Attiva o disattiva se l'SDK invierà richieste agli endpoint dual-stack. Per ulteriori informazioni sugli endpoint dual-stack, che supportano sia il traffico IPv4 che IPv6, consulta [Using Amazon S3 dual-stack endpoint nella Amazon Simple Storage Service User Guide](#). Gli endpoint dual-stack sono disponibili per alcuni servizi in alcune regioni.

Valore predefinito: `false`

Valori validi:

- **true**— L'SDK o lo strumento tenteranno di utilizzare gli endpoint dual-stack per effettuare richieste di rete. Se non esiste un endpoint dual-stack per il servizio e/o, la richiesta avrà esito negativo. Regione AWS
- **false**— L'SDK o lo strumento non utilizzeranno endpoint dual-stack per effettuare richieste di rete.

**use\_fips\_endpoint**- impostazione di AWS **config** file condivisi, **AWS\_USE\_FIPS\_ENDPOINT**- variabile d'ambiente, **aws.useFipsEndpoint**- Proprietà del sistema JVM: solo Java/Kotlin

Attiva o disattiva se l'SDK o lo strumento invieranno richieste agli endpoint conformi a FIPS. I Federal Information Processing Standards (FIPS) sono un insieme di requisiti di sicurezza del governo degli Stati Uniti per i dati e la loro crittografia. Le agenzie governative, i partner e coloro che desiderano fare affari con il governo federale sono tenuti a rispettare le linee guida FIPS. A differenza degli AWS endpoint standard, gli endpoint FIPS utilizzano una libreria software TLS conforme a FIPS 140-2. Se questa impostazione è abilitata e non esiste un endpoint FIPS per il servizio in uso, la chiamata potrebbe non riuscire. Regione AWS [Endpoint specifici del servizio](#) e l' `--endpoint-url` opzione per AWS Command Line Interface sovrascrivere questa impostazione.

Per ulteriori informazioni su altri modi per specificare gli endpoint FIPS tramite Regione AWS, consulta [FIPS Endpoints by Service](#). Per ulteriori informazioni sugli endpoint del servizio Amazon Elastic Compute Cloud, consulta gli endpoint [Dual-stack \(IPv4 e IPv6\)](#) nell'Amazon EC2 API Reference.

Valore predefinito: `false`

Valori validi:

- **true**— L'SDK o lo strumento invieranno le richieste agli endpoint conformi a FIPS.
- **false**— L'SDK o lo strumento non invieranno richieste agli endpoint conformi a FIPS.

## Compatibilità con gli AWS SDK

I seguenti SDK supportano le funzionalità e le impostazioni descritte in questo argomento. Vengono annotate eventuali eccezioni parziali. Tutte le impostazioni delle proprietà del sistema JVM sono supportate solo da AWS SDK for Java and the. SDK AWS for Kotlin

SDK	Sì	Note o ulteriori informazioni
<a href="#">AWS CLI v2</a>	Sì	
<a href="#">SDK per C++</a>	Sì	
<a href="#">SDK per Go V2 (1.x)</a>	Sì	

SDK	Sì o	Note o ulteriori informazioni
<a href="#">SDK per Go 1.x (V1)</a>	Sì	<a href="#">Per utilizzare le impostazioni dei config file condivisi, devi attivare il caricamento dal file di configurazione; vedi Sessioni.</a>
<a href="#">SDK per Java 2.x</a>	Sì	
<a href="#">SDK per Java 1.x</a>	No	
<a href="#">SDK per 3.x JavaScript</a>	Sì	
<a href="#">SDK per 2.x JavaScript</a>	Sì	
<a href="#">SDK per Kotlin</a>	Sì	
<a href="#">SDK per .NET 3.x</a>	Sì	
<a href="#">SDK per PHP 3.x</a>	Sì	
<a href="#">SDK per Python (Boto3)</a>	Sì	
<a href="#">SDK per Ruby 3.x</a>	Sì	
<a href="#">SDK per Rust</a>	Sì	
<a href="#">Strumenti per PowerShell</a>	Sì	

## Rilevamento di endpoint

Gli SDK utilizzano l'endpoint discovery per accedere agli endpoint dei servizi (URL per accedere a varie risorse), pur mantenendo la flessibilità necessaria per modificare gli URL in base AWS alle esigenze. In questo modo, il codice può rilevare automaticamente nuovi endpoint. Non esistono endpoint fissi per alcuni servizi. Al contrario, è possibile ottenere gli endpoint disponibili durante il runtime effettuando prima una richiesta per ottenere gli endpoint. Dopo aver recuperato gli endpoint disponibili, il codice utilizza quindi l'endpoint per accedere ad altre operazioni. Ad esempio, per Amazon Timestream, l'SDK effettua `DescribeEndpoints` una richiesta per recuperare gli endpoint disponibili e quindi utilizza tali endpoint per completare operazioni specifiche come `CreateDatabase` `CreateTable`

L'individuazione degli endpoint è obbligatoria in alcuni servizi e facoltativa in altri. L'impostazione predefinita è uno dei due `true` o `false` dipende dal fatto che il servizio richieda o meno il rilevamento degli endpoint. Ad esempio, per impostazione predefinita Timestream è impostato su `true` Amazon DynamoDB per impostazione predefinita è `false`. Per i servizi in cui non è richiesto il rilevamento degli endpoint, il rilevamento degli endpoint non è abilitato. Le opzioni di configurazione sono invece disponibili tramite variabili di ambiente, AWS `config` file condiviso o costrutti di codice SDK (ad esempio, classi di configurazione). Per le operazioni in cui è richiesto il rilevamento degli endpoint, l'SDK tenta automaticamente di scoprire un endpoint.

Configura questa funzionalità utilizzando quanto segue:

**endpoint\_discovery\_enabled**- impostazione dei AWS `config` file condivisi, **AWS\_ENABLE\_ENDPOINT\_DISCOVERY**- variabile d'ambiente, **aws.endpointDiscoveryEnabled**- Proprietà del sistema JVM: solo Java/Kotlin, Per configurare il valore direttamente nel codice, consulta direttamente il tuo SDK specifico.

Abilita/disabilita il rilevamento degli endpoint per i servizi in cui l'individuazione degli endpoint è facoltativa. L'individuazione degli endpoint è richiesta in alcuni servizi.

Valore predefinito: `false`

Valori validi:

- **true**— L'SDK dovrebbe tentare automaticamente di individuare un endpoint per i servizi in cui l'individuazione degli endpoint è facoltativa.
- **false**— L'SDK non dovrebbe tentare automaticamente di rilevare un endpoint per servizi in cui l'individuazione degli endpoint è facoltativa.

## Compatibilità con gli SDK AWS

I seguenti SDK supportano le funzionalità e le impostazioni descritte in questo argomento. Vengono annotate eventuali eccezioni parziali. Tutte le impostazioni delle proprietà del sistema JVM sono supportate solo da AWS SDK for Java and the. SDK AWS for Kotlin

SDK	Sì	Note o ulteriori informazioni
<a href="#">AWS CLI v2</a>	Sì	

SDK	Sì o	Note o ulteriori informazioni
<a href="#">SDK per C++</a>	Sì	
<a href="#">SDK per Go V2 (1.x)</a>	Sì	
<a href="#">SDK per Go 1.x (V1)</a>	Sì	<a href="#">Per utilizzare le impostazioni dei config file condivisi, devi attivare il caricamento dal file di configurazione; vedi Sessioni.</a>
<a href="#">SDK per Java 2.x</a>	Sì	L'SDK for Java 2.x lo <code>AWS_ENDPOINT_DISCOVERY_ENABLED</code> utilizza per il nome della variabile di ambiente.
<a href="#">SDK per Java 1.x</a>	Sì	
<a href="#">SDK per 3.x JavaScript</a>	Sì	
<a href="#">SDK per 2.x JavaScript</a>	Sì	
<a href="#">SDK per Kotlin</a>	Sì	
<a href="#">SDK per .NET 3.x</a>	Sì	
<a href="#">SDK per PHP 3.x</a>	Sì	
<a href="#">SDK per Python (Boto3)</a>	Sì	
<a href="#">SDK per Ruby 3.x</a>	Sì	
<a href="#">SDK per Rust</a>	Parzia	Supportato solo per Timestream.
<a href="#">Strumenti per PowerShell</a>	Sì	

## Impostazioni generali di configurazione

Gli SDK supportano alcune impostazioni generali che configurano i comportamenti complessivi dell'SDK.

Configura questa funzionalità utilizzando quanto segue:



## **api\_versions**- impostazione dei AWS **config** file condivisi

Alcuni AWS servizi mantengono più versioni dell'API per supportare la compatibilità con le versioni precedenti. Per impostazione predefinita, SDK e AWS CLI operazioni utilizzano l'ultima versione API disponibile. Per richiedere una versione API specifica da utilizzare per le tue richieste, includi l'`api_versions` impostazione nel tuo profilo.

Valore predefinito: Nessuno. (L'ultima versione dell'API viene utilizzata dall'SDK.)

Valori validi: si tratta di un'impostazione annidata seguita da una o più righe rientrate, ciascuna delle quali identifica un AWS servizio e la versione dell'API da utilizzare. Consulta la documentazione del AWS servizio per capire quali versioni dell'API sono disponibili.

L'esempio imposta una versione API specifica per due AWS servizi nel `config` file. Queste versioni API vengono utilizzate solo per i comandi eseguiti nel profilo che contiene queste impostazioni. I comandi per qualsiasi altro servizio utilizzano la versione più recente dell'API di quel servizio.

```
api_versions =  
  ec2 = 2015-03-01  
  cloudfront = 2015-09-017
```

## **ca\_bundle**- impostazione di AWS **config** file condivisi, **AWS\_CA\_BUNDLE**- variabile d'ambiente

Specifica il percorso di un pacchetto di certificati personalizzato (un file con `.pem` estensione) da utilizzare per stabilire connessioni SSL/TLS.

Valore predefinito: nessuno

Valori validi: specificate il percorso completo o un nome di file di base. Se è presente un nome di file di base, il sistema tenta di trovare il programma all'interno delle cartelle specificate dalla variabile di `PATH` ambiente.

Esempio di impostazione di questo valore nel `config` file:

```
[default]  
ca_bundle = dev/apps/ca-certs/cabundle-2019mar05.pem
```

Esempio in Linux/macOS di impostazione delle variabili di ambiente tramite riga di comando:

```
export AWS_CA_BUNDLE=/dev/apps/ca-certs/cabundle-2019mar05.pem
```

Esempio in Windows di impostazione delle variabili di ambiente tramite riga di comando:

```
setx AWS_CA_BUNDLE C:\dev\apps\ca-certs\cabundle-2019mar05.pem
```

**output-** impostazione di AWS **config** file condivisi

Specifica il modo in cui i risultati vengono formattati negli AWS SDK AWS CLI e negli altri strumenti.

Valore predefinito: `json`

Valori validi:

- **json**— [L'output è formattato come stringa JSON.](#)
- **yaml**— [L'output è formattato come stringa YAML.](#)
- **yaml-stream**— [L'output viene trasmesso in streaming e formattato come stringa YAML.](#) Lo streaming consente una gestione più rapida di tipi di dati di grandi dimensioni.
- **text**— L'output è formattato come più righe di valori di stringa separati da tabulazioni. Questo può essere utile per passare l'output a un elaboratore di testi, ad esempio `grep`, `sed` o `awk`
- **table**— L'output viene formattato come tabella utilizzando i caratteri `+`/`-` per formare i bordi delle celle. In genere presenta le informazioni in un formato comprensibile molto più semplice da leggere rispetto ad altri, ma non altrettanto utile a livello programmatico.

**parameter\_validation-** impostazione di file condivisi AWS **config**

Specifica se l'SDK o lo strumento tenta di convalidare i parametri della riga di comando prima di inviarli all'endpoint del AWS servizio.

Valore predefinito: `true`

Valori validi:

- **true** – Il valore predefinito. L'SDK o lo strumento esegue la convalida lato client dei parametri della riga di comando. Ciò consente all'SDK o allo strumento di confermare la validità dei parametri e rileva alcuni errori. L'SDK o lo strumento possono rifiutare le richieste non valide prima di inviarle all'endpoint del servizio. AWS
- **false**— L'SDK o lo strumento non convalidano i parametri della riga di comando prima di inviarli all'endpoint del servizio. AWS L'endpoint del AWS servizio è responsabile della convalida di tutte le richieste e del rifiuto delle richieste non valide.

## Compatibilità con gli SDK AWS

I seguenti SDK supportano le funzionalità e le impostazioni descritte in questo argomento. Vengono annotate eventuali eccezioni parziali. Tutte le impostazioni delle proprietà del sistema JVM sono supportate solo da AWS SDK for Java and the. SDK AWS for Kotlin

SDK	Sì o	Note o ulteriori informazioni
<a href="#">AWS CLI v2</a>	Parziale	<code>api_versions</code> non supportato.
<a href="#">SDK per C++</a>	Sì	
<a href="#">SDK per Go V2 (1.x)</a>	Parziale	<code>api_versions</code> e <code>parameter_validation</code> non supportato.
<a href="#">SDK per Go 1.x (V1)</a>	Parziale	<code>api_versions</code> e <code>parameter_validation</code> non supportato. Per utilizzare le impostazioni dei config file condivisi, devi attivare il caricamento dal file di configurazione; vedi <a href="#">Sessioni</a> .
<a href="#">SDK per Java 2.x</a>	No	
<a href="#">SDK per Java 1.x</a>	No	
<a href="#">SDK per 3.x JavaScript</a>	Sì	
<a href="#">SDK per 2.x JavaScript</a>	Sì	
<a href="#">SDK per Kotlin</a>	No	
<a href="#">SDK per .NET 3.x</a>	No	
<a href="#">SDK per PHP 3.x</a>	Sì	
<a href="#">SDK per Python (Boto3)</a>	Sì	
<a href="#">SDK per Ruby 3.x</a>	Sì	
<a href="#">SDK per Rust</a>	No	

SDK	Sì	Note o ulteriori informazioni
<a href="#">Strumenti per PowerShell</a>	No	

## Cliente IMDS

Gli SDK implementano un client Instance Metadata Service Version 2 (IMDSv2) utilizzando richieste orientate alla sessione. Per ulteriori informazioni su IMDSv2, consulta [Use IMDSv2 nella Amazon EC2 User Guide for Linux Instances](#) o Use [IMDSv2 nella Amazon EC2 User Guide for Windows Instances](#). Il client IMDS è configurabile tramite un oggetto di configurazione del client disponibile nella base di codice SDK.

Configura questa funzionalità utilizzando quanto segue:

**retries**- membro dell'oggetto di configurazione del client

Il numero di tentativi di riprova aggiuntivi per ogni richiesta non riuscita.

Valore predefinito: 3

Valori validi: numero maggiore di 0.

**port**- membro dell'oggetto di configurazione del client

La porta per l'endpoint.

Valore predefinito: 80

Valori validi: Numero.

**token\_ttl**- membro dell'oggetto di configurazione del client

Il TTL del token.

Valore predefinito: 21.600 secondi (6 ore, il tempo massimo assegnato).

Valori validi: Numero.

**endpoint**- membro dell'oggetto di configurazione del client

L'endpoint di IMDS.

Valore predefinito: se è `endpoint_mode` uguale `IPv4`, l'endpoint predefinito è.

`http://169.254.169.254` Se è `endpoint_mode` uguale, l'endpoint predefinito è. `IPv6`

`http://[fd00:ec2::254]`

Valori validi: URI valido.

Le seguenti opzioni sono supportate dalla maggior parte degli SDK. Consulta la tua base di codice SDK specifica per i dettagli.

**`endpoint_mode`**- membro dell'oggetto di configurazione del client

La modalità endpoint di IMDS.

Valore predefinito: `IPv4`

Valori validi: `IPv4`, `IPv6`

**`http_open_timeout`**- membro dell'oggetto di configurazione del client (il nome può variare)

Il numero di secondi di attesa per l'apertura della connessione.

Valore predefinito: 1 secondo.

Valori validi: numero maggiore di 0.

**`http_read_timeout`**- membro dell'oggetto di configurazione del client (il nome può variare)

Il numero di secondi per la lettura di un blocco di dati.

Valore predefinito: 1 secondo.

Valori validi: numero maggiore di 0.

**`http_debug_output`**- membro dell'oggetto di configurazione del client (il nome può variare)

Imposta un flusso di output per il debug.

Valore predefinito: Nessuno.

Valori validi: un flusso di I/O valido, come `STDOUT`.

**`backoff`**- membro dell'oggetto di configurazione del client (il nome può variare)

Il numero di secondi trascorsi a dormire tra un tentativo e l'altro o la funzione di backoff fornita dal cliente per effettuare una chiamata. Ciò ha la precedenza sulla strategia di backoff esponenziale predefinita.

Valore predefinito: varia in base all'SDK.

Valori validi: varia in base all'SDK. Può essere un valore numerico o una chiamata a una funzione personalizzata.

## Compatibilità con AWS gli SDK

I seguenti SDK supportano le funzionalità e le impostazioni descritte in questo argomento. Vengono annotate eventuali eccezioni parziali. Tutte le impostazioni delle proprietà del sistema JVM sono supportate solo da AWS SDK for Java and the. SDK AWS for Kotlin

SDK	Sì o No	Note o ulteriori informazioni
<a href="#">AWS CLI v2</a>	Sì	
<a href="#">SDK per C++</a>	No	IMDSv2 utilizzato solo internamente. Per informazioni, consulta <a href="#">Provider di credenziali IMDS</a> .
<a href="#">SDK per Go V2 (1.x)</a>	Sì	
<a href="#">SDK per Go 1.x (V1)</a>	Sì	
<a href="#">SDK per Java 2.x</a>	Sì	
<a href="#">SDK per Java 1.x</a>	Sì	
<a href="#">SDK per 3.x JavaScript</a>	Sì	
<a href="#">SDK per 2.x JavaScript</a>	Sì	
<a href="#">SDK per Kotlin</a>	Sì	
<a href="#">SDK per .NET 3.x</a>	Sì	
<a href="#">SDK per PHP 3.x</a>	Sì	
<a href="#">SDK per Python (Boto3)</a>	Sì	
<a href="#">SDK per Ruby 3.x</a>	Sì	

SDK	Sì	Note o ulteriori informazioni
<a href="#">SDK per Rust</a>	Sì	
<a href="#">Strumenti per PowerShell</a>	Sì	

## Comportamento di ripetizione

Il comportamento Riprova include le impostazioni relative al modo in cui gli SDK tentano di ripristinare gli errori derivanti dalle richieste effettuate a. Servizi AWS

Configura questa funzionalità utilizzando quanto segue:

**max\_attempts**- impostazione dei file **AWS config** condivisi, **AWS\_MAX\_ATTEMPTS**- variabile d'ambiente, **aws.maxAttempts**- Proprietà del sistema JVM: solo Java/Kotlin

Specifica il numero massimo di tentativi da effettuare su una richiesta.

Valore predefinito: se questo valore non è specificato, il valore predefinito dipende dal valore dell'`retry_mode` impostazione:

- Se lo `retry_mode` è `legacy`: utilizza un valore predefinito specifico per il tuo SDK (consulta la guida SDK specifica o la base di codice dell'SDK per `max_attempts` i valori predefiniti).
- Se lo `retry_mode` è `standard`: effettua tre tentativi.
- Se lo `retry_mode` è `adaptive`: effettua tre tentativi.

Valori validi: numero maggiore di 0.

**retry\_mode**- impostazione condivisa **AWS config** dei file, **AWS\_RETRY\_MODE**- variabile d'ambiente, **aws.retryMode**- Proprietà del sistema JVM: solo Java/Kotlin

Specifica in che modo l'SDK o lo strumento di sviluppo tenta di riprovare.

Valore predefinito: `legacy` è la strategia di riprova predefinita.

Valori validi:

- `legacy`— Specifico per il tuo SDK (consulta la tua guida SDK specifica o la base di codice dell'SDK).

- **standard**— Il set standard di regole di riprova tra gli SDK. AWS Questa modalità include un set standard di errori che vengono ripetuti e il supporto per le quote di tentativi. Il numero massimo predefinito di tentativi con questa modalità è tre, a meno che non `max_attempts` sia configurata in modo esplicito.
- **adaptive**— Una modalità sperimentale di ripetizione che include le funzionalità della modalità standard ma include la limitazione automatica sul lato client. Poiché questa modalità è sperimentale, potrebbe modificare il comportamento in futuro.

Di seguito è riportato lo pseudocodice di alto livello per entrambe le `standard` modalità e `retry: adaptive`

```
MakeSDKRequest() {
  attempts = 0
  loop {
    GetSendToken()
    response = SendHTTPRequest()
    RequestBookkeeping(response)
    if not Retryable(response)
      return response
    attempts += 1
    if attempts >= MAX_ATTEMPTS:
      return response
    if not HasRetryQuota(response)
      return response
    delay = ExponentialBackoff(attempts)
    sleep(delay)
  }
}
```

Di seguito sono riportati ulteriori dettagli sui componenti utilizzati nello pseudocodice:

### **GetSendToken:**

I token bucket vengono utilizzati solo in modalità riprova. `adaptive` I token bucket impongono una frequenza massima di richieste richiedendo la disponibilità di un token per avviare una richiesta. Il client SDK è configurabile in modo da fallire rapidamente la richiesta o bloccarla finché non diventa disponibile un token.

Client Side Rate Limiting è un algoritmo che inizialmente consente di effettuare richieste in qualsiasi momento, fino al limite consentito dal token. Tuttavia, dopo che viene rilevata una risposta limitata,



il client `rate-of-request` viene limitato di conseguenza. La soglia di token viene inoltre aumentata di conseguenza se vengono ricevute risposte positive.

Grazie alla limitazione adattiva della velocità, gli SDK possono rallentare la velocità di invio delle richieste per soddisfare meglio la capacità di. Servizi AWS

### **SendHTTPRequest:**

La maggior parte degli AWS SDK utilizza una libreria HTTP che utilizza pool di connessioni in modo da poter riutilizzare una connessione esistente quando si effettua una richiesta HTTP. In genere, le connessioni vengono riutilizzate quando si riprovano le richieste a causa di errori di limitazione. Le richieste non vengono riutilizzate quando si riprova a causa di errori temporanei.

### **RequestBookkeeping:**

La quota di tentativi deve essere aggiornata se la richiesta ha esito positivo. Solo per la modalità `adaptive` riprova, la variabile di stato `maxsendrate` viene aggiornata in base al tipo di risposta ricevuta.

### **Retryable:**

Questo passaggio determina se una risposta può essere ritentata in base a quanto segue:

- Codice di stato HTTP .
- Il codice di errore restituito dal servizio.
- Errori di connessione, definiti come qualsiasi errore ricevuto dall'SDK in cui non viene ricevuta una risposta HTTP dal servizio.

Gli errori transitori (codici di stato HTTP 400, 408, 500, 502, 503 e 504) e gli errori di limitazione (codici di stato HTTP 400, 403, 429, 502, 503 e 509) possono essere tutti potenzialmente ritentati. Il comportamento dei nuovi tentativi dell'SDK viene determinato in combinazione con codici di errore o altri dati del servizio.

### **MAX\_ATTEMPTS:**

Specificato dall'impostazione del `config` file o dalla variabile di ambiente.

### **HasRetryQuota**

Questo passaggio limita le richieste di nuovi tentativi richiedendo che un token sia disponibile nel bucket delle quote di tentativi. I gruppi di quote per nuovi tentativi sono un meccanismo per

impedire nuovi tentativi che difficilmente avranno successo. Queste quote dipendono dall'SDK, spesso dipendono dal client e talvolta dipendono anche dagli endpoint del servizio. I token di quota di tentativi disponibili vengono rimossi quando le richieste hanno esito negativo per vari motivi e ripristinati quando hanno esito positivo. Quando non rimangono token, il ciclo di ripetizione dei tentativi viene chiuso.

## ExponentialBackoff

Per un errore che può essere riprovato, il ritardo tra i tentativi viene calcolato utilizzando un backoff esponenziale troncato. Gli SDK utilizzano un backoff esponenziale binario troncato con jitter.

L'algoritmo seguente mostra come viene definita la quantità di tempo per dormire, in secondi, per una risposta a una richiesta:  $i$

$$\text{seconds\_to\_sleep\_i} = \min(b * r^i, \text{MAX\_BACKOFF})$$

Nell'algoritmo precedente, si applicano i seguenti valori:

$b$  = random number within the range of:  $0 \leq b \leq 1$

$r = 2$

$\text{MAX\_BACKOFF} = 20$  seconds per la maggior parte degli SDK. Per conferma, consulta la guida SDK o il codice sorgente specifici.

## Compatibilità con AWS gli SDK

I seguenti SDK supportano le funzionalità e le impostazioni descritte in questo argomento. Vengono annotate eventuali eccezioni parziali. Tutte le impostazioni delle proprietà del sistema JVM sono supportate solo da AWS SDK for Java and the. SDK AWS for Kotlin

SDK	Sì o	Note o ulteriori informazioni
<a href="#">AWS CLI v2</a>	Sì	
<a href="#">SDK per C++</a>	Sì	
<a href="#">SDK per Go V2 (1.x)</a>	Sì	
<a href="#">SDK per Go 1.x (V1)</a>	No	

SDK	Sì o	Note o ulteriori informazioni
<a href="#">SDK per Java 2.x</a>	Sì	
<a href="#">SDK per Java 1.x</a>	Sì	
<a href="#">SDK per 3.x JavaScript</a>	Sì	
<a href="#">SDK per 2.x JavaScript</a>	No	Supporta un numero massimo di tentativi, un backoff esponenziale con jitter e un'opzione per un metodo personalizzato per il backoff dei tentativi.
<a href="#">SDK per Kotlin</a>	Sì	
<a href="#">SDK per .NET 3.x</a>	Sì	
<a href="#">SDK per PHP 3.x</a>	Sì	
<a href="#">SDK per Python (Boto3)</a>	Sì	
<a href="#">SDK per Ruby 3.x</a>	Sì	
<a href="#">SDK per Rust</a>	Sì	
<a href="#">Strumenti per PowerShell</a>	Sì	

## Richiesta di compressione

AWS Gli SDK e gli strumenti possono comprimere automaticamente i payload quando si inviano richieste al supporto che riceve payload compressi. Servizi AWS La compressione del payload sul client prima di inviarlo a un servizio può ridurre il numero complessivo di richieste e la larghezza di banda necessari per inviare dati al servizio, nonché ridurre le richieste non riuscite a causa delle limitazioni del servizio sulla dimensione del payload. Per la compressione, l'SDK o lo strumento seleziona un algoritmo di codifica supportato sia dal servizio che dall'SDK. Tuttavia, l'elenco attuale delle possibili codifiche è costituito solo da gzip, ma potrebbe espandersi in futuro.

La compressione delle richieste può essere particolarmente utile se l'applicazione utilizza [Amazon CloudWatch](#). CloudWatch è un servizio di monitoraggio e osservabilità che raccoglie dati operativi e

di monitoraggio sotto forma di log, metriche ed eventi. [Un esempio di funzionamento di servizio che supporta la compressione è CloudWatch il metodo API. PutMetricData](#)

Configura questa funzionalità utilizzando quanto segue:

**disable\_request\_compression**- impostazione dei AWS **config** file condivisi, **AWS\_DISABLE\_REQUEST\_COMPRESSION**- variabile d'ambiente, **aws.disableRequestCompression**- Proprietà del sistema JVM: solo Java/Kotlin

Attiva o disattiva se l'SDK o lo strumento comprimeranno un payload prima di inviare una richiesta.

Valore predefinito: `false`

Valori validi:

- **true**— Disattiva la compressione delle richieste.
- **false**— Usa la compressione delle richieste quando possibile.

**request\_min\_compression\_size\_bytes**- impostazione dei AWS **config** file condivisi, **AWS\_REQUEST\_MIN\_COMPRESSION\_SIZE\_BYTES**- variabile d'ambiente, **aws.requestMinCompressionSizeBytes**- Proprietà del sistema JVM: solo Java/Kotlin

Imposta la dimensione minima in byte del corpo della richiesta che l'SDK o lo strumento devono comprimere. I carichi utili di piccole dimensioni possono allungarsi quando vengono compressi, quindi esiste un limite inferiore in base al quale è opportuno eseguire la compressione. Questo valore è inclusivo, viene compressa una dimensione della richiesta maggiore o uguale al valore.

Valore predefinito: 10240 byte

Valori validi: valore intero compreso tra 0 e 10485760 byte inclusi.

## AWS Compatibilità con gli SDK

I seguenti SDK supportano le funzionalità e le impostazioni descritte in questo argomento. Vengono annotate eventuali eccezioni parziali. Tutte le impostazioni delle proprietà del sistema JVM sono supportate solo da AWS SDK for Java and the. SDK AWS for Kotlin

SDK	Sì o No	Note o ulteriori informazioni
<a href="#">AWS CLI v2</a>	Sì	
<a href="#">SDK per C++</a>	Sì	
<a href="#">SDK per Go V2 (1.x)</a>	Sì	
<a href="#">SDK per Go 1.x (V1)</a>	No	
<a href="#">SDK per Java 2.x</a>	Sì	
<a href="#">SDK per Java 1.x</a>	No	
<a href="#">SDK per 3.x JavaScript</a>	Sì	
<a href="#">SDK per 2.x JavaScript</a>	No	
<a href="#">SDK per Kotlin</a>	Sì	
<a href="#">SDK per .NET 3.x</a>	Sì	
<a href="#">SDK per PHP 3.x</a>	Sì	
<a href="#">SDK per Python (Boto3)</a>	Sì	
<a href="#">SDK per Ruby 3.x</a>	Sì	
<a href="#">SDK per Rust</a>	No	
<a href="#">Strumenti per PowerShell</a>	Sì	

## Endpoint specifici del servizio

La configurazione degli endpoint specifica per il servizio offre la possibilità di utilizzare un endpoint di propria scelta per le richieste API e di mantenere tale scelta. Queste impostazioni offrono la flessibilità necessaria per supportare endpoint locali, endpoint VPC e ambienti di sviluppo locale di terze parti AWS. È possibile utilizzare endpoint diversi per ambienti di test e produzione. È possibile specificare un URL di endpoint per singoli utenti. Servizi AWS

Configura questa funzionalità utilizzando quanto segue:

**endpoint\_url**- impostazione dei AWS **config** file condivisi, **AWS\_ENDPOINT\_URL**- variabile d'ambiente, **aws.endpointUrl**- Proprietà del sistema JVM: solo Java/Kotlin

Se specificata direttamente all'interno di un profilo o come variabile di ambiente, questa impostazione specifica l'endpoint utilizzato per tutte le richieste di servizio. Questo endpoint viene sovrascritto da qualsiasi endpoint configurato specifico del servizio.

È inoltre possibile utilizzare questa impostazione all'interno di una **services** sezione di un AWS **config** file condiviso per impostare un endpoint personalizzato per un servizio specifico. Per un elenco di tutte le chiavi identificative del servizio da utilizzare per le sottosezioni all'interno della **services** sezione, vedere. [Identificatori per endpoint specifici del servizio](#)

Valore predefinito: none

Valori validi: un URL che include lo schema e l'host per l'endpoint. L'URL può facoltativamente contenere un componente del percorso che contiene uno o più segmenti di percorso.

**AWS\_ENDPOINT\_URL\_<SERVICE>**- variabile di ambiente, **aws.endpointUrl<ServiceName>**- Proprietà del sistema JVM: solo Java/Kotlin

**AWS\_ENDPOINT\_URL\_<SERVICE>**, **<SERVICE>** dov'è l' Servizio AWS identificatore, imposta un endpoint personalizzato per un servizio specifico. Per un elenco di tutte le variabili di ambiente specifiche del servizio, vedere. [Identificatori per endpoint specifici del servizio](#)

Questo endpoint specifico del servizio sostituisce qualsiasi endpoint globale impostato.

**AWS\_ENDPOINT\_URL**

Valore predefinito: none

Valori validi: un URL che include lo schema e l'host per l'endpoint. L'URL può facoltativamente contenere un componente del percorso che contiene uno o più segmenti di percorso.

**ignore\_configured\_endpoint\_urls**- impostazione condivisa AWS **config** dei file, **AWS\_IGNORE\_CONFIGURED\_ENDPOINT\_URLS**- variabile d'ambiente, **aws.ignoreConfiguredEndpointUrls**- Proprietà del sistema JVM: solo Java/Kotlin

Questa impostazione viene utilizzata per ignorare tutte le configurazioni personalizzate degli endpoint.

Tieni presente che qualsiasi endpoint esplicito impostato nel codice o su uno stesso client di servizio viene utilizzato indipendentemente da questa impostazione. Ad esempio, l'inclusione del

parametro della riga di `--endpoint-url` comando con un AWS CLI comando o il passaggio di un URL di endpoint a un costruttore del client avrà sempre effetto.

Valore predefinito: `false`

Valori validi:

- **true**— L'SDK o lo strumento non leggono alcuna opzione di configurazione personalizzata dal `config` file condiviso o dalle variabili di ambiente per l'impostazione di un URL dell'endpoint.
- **false**— L'SDK o lo strumento utilizza tutti gli endpoint disponibili forniti dall'utente dal `config` file condiviso o dalle variabili di ambiente.

## Configura gli endpoint utilizzando variabili di ambiente

Per indirizzare le richieste di tutti i servizi a un URL endpoint personalizzato, imposta la variabile di ambiente `AWS_ENDPOINT_URL` globale.

```
export AWS_ENDPOINT_URL=http://localhost:4567
```

Per indirizzare le richieste per un URL di endpoint specifico Servizio AWS a un URL di endpoint personalizzato, utilizza la variabile di `AWS_ENDPOINT_URL_<SERVICE>` ambiente. Amazon DynamoDB ha un `serviceId`. [DynamoDB](#) Per questo servizio, la variabile di ambiente dell'URL dell'endpoint è `AWS_ENDPOINT_URL_DYNAMODB`. Questo endpoint ha la precedenza sull'endpoint globale impostato per questo servizio. `AWS_ENDPOINT_URL`

```
export AWS_ENDPOINT_URL_DYNAMODB=http://localhost:5678
```

Come altro esempio, AWS Elastic Beanstalk ha un `di.serviceId` [Elastic Beanstalk](#) L' Servizio AWS identificatore si basa sul modello API `serviceId` sostituendo tutti gli spazi con caratteri di sottolineatura e tutte le lettere maiuscole. Per impostare l'endpoint per questo servizio, la variabile di ambiente corrispondente è `AWS_ENDPOINT_URL_ELASTIC_BEANSTALK` Per un elenco di tutte le variabili di ambiente specifiche del servizio, vedere. [Identificatori per endpoint specifici del servizio](#)

```
export AWS_ENDPOINT_URL_ELASTIC_BEANSTALK=http://localhost:5567
```

## Configurare gli endpoint utilizzando il file condiviso **config**

Nel `config` file condiviso, `endpoint_url` viene utilizzato in luoghi diversi per funzionalità diverse.

- `endpoint_url` specificato direttamente all'interno di `a profile` rende quell'endpoint l'endpoint globale.
- `endpoint_url` annidato sotto una chiave identificativa del servizio all'interno di una `services` sezione fa sì che l'endpoint si applichi alle richieste fatte solo a quel servizio. Per i dettagli sulla definizione di una `services` sezione nel config file condiviso, consulta. [Formato del file di configurazione](#)

L'esempio seguente utilizza una `services` definizione per configurare un URL di endpoint specifico del servizio da utilizzare per Amazon S3 e un endpoint globale personalizzato da utilizzare per tutti gli altri servizi:

```
[profile dev-s3-specific-and-global]
endpoint_url = http://localhost:1234
services = s3-specific

[services s3-specific]
s3 =
  endpoint_url = https://play.min.io:9000
```

Un singolo profilo può configurare gli endpoint per più servizi. Questo esempio mostra come impostare gli URL degli endpoint specifici del servizio per Amazon S3 e nello stesso profilo. AWS Elastic Beanstalk AWS Elastic Beanstalk ha un. `serviceId` [Elastic Beanstalk](#) L' Servizio AWS identificatore si basa sul modello API `serviceId` sostituendo tutti gli spazi con caratteri di sottolineatura e tutte le lettere minuscole. Pertanto, la chiave identificativa del servizio diventa `elastic_beanstalk` e le impostazioni per questo servizio iniziano sulla linea. `elastic_beanstalk =` Per un elenco di tutte le chiavi identificative del servizio da utilizzare nella `services` sezione, vedere. [Identificatori per endpoint specifici del servizio](#)

```
[services testing-s3-and-eb]
s3 =
  endpoint_url = http://localhost:4567
elastic_beanstalk =
  endpoint_url = http://localhost:8000

[profile dev]
services = testing-s3-and-eb
```

La sezione di configurazione del servizio può essere utilizzata da più profili. Ad esempio, due profili possono utilizzare la stessa `services` definizione modificando altre proprietà del profilo:



```
[services testing-s3]  
s3 =  
    endpoint_url = https://localhost:4567  
  
[profile testing-json]  
output = json  
services = testing-s3  
  
[profile testing-text]  
output = text  
services = testing-s3
```

## Configura gli endpoint nei profili utilizzando credenziali basate sui ruoli

Se il tuo profilo ha credenziali basate sui ruoli configurate tramite un `source_profile` parametro per IAM Assume Role Functionality, l'SDK utilizza solo le configurazioni di servizio per il profilo specificato. Non utilizza profili concatenati a ruoli. Ad esempio, utilizzando il seguente config file condiviso:

```
[profile A]  
credential_source = Ec2InstanceMetadata  
endpoint_url = https://profile-a-endpoint.aws/  
  
[profile B]  
source_profile = A  
role_arn = arn:aws:iam::123456789012:role/roleB  
services = profileB  
  
[services profileB]  
ec2 =  
    endpoint_url = https://profile-b-ec2-endpoint.aws
```

Se usi il profilo B ed effettui una chiamata nel codice verso Amazon EC2, l'endpoint si risolve come `https://profile-b-ec2-endpoint.aws`. Se il codice invia una richiesta a qualsiasi altro servizio, la risoluzione dell'endpoint non seguirà alcuna logica personalizzata. L'endpoint non si risolve nell'endpoint globale definito nel profilo. A Affinché un endpoint globale abbia effetto sul profiloB, è necessario `endpoint_url` impostarlo direttamente all'interno del profilo. B Per ulteriori informazioni in merito all'impostazione `source_profile`, consulta [Assumi il ruolo di fornitore di credenziali](#).

## Precedenza delle impostazioni

Le impostazioni di questa funzionalità possono essere utilizzate contemporaneamente, ma solo un valore avrà la priorità per servizio. Per le chiamate API effettuate verso un determinato valore Servizio AWS, viene utilizzato il seguente ordine per selezionare un valore:

1. Qualsiasi impostazione esplicita impostata nel codice o su un client di servizio stesso ha la precedenza su qualsiasi altra cosa.
  - Per il AWS CLI, questo è il valore fornito dal parametro della `--endpoint-url` riga di comando. Per un SDK, le assegnazioni esplicite possono assumere la forma di un parametro impostato quando si crea un'istanza di un client o di un Servizio AWS oggetto di configurazione.
2. Il valore fornito da una variabile di ambiente specifica del servizio, ad esempio. `AWS_ENDPOINT_URL_DYNAMODB`
3. Il valore fornito dalla variabile di ambiente `AWS_ENDPOINT_URL` globale dell'endpoint.
4. Il valore fornito dall'`endpoint_url` impostazione annidata in una chiave di identificazione del servizio all'interno di una `services` sezione del file condiviso. `config`
5. Il valore fornito dall'`endpoint_url` impostazione specificato direttamente all'interno `profile` di uno dei file condivisi `config`.
6. Qualsiasi URL di endpoint predefinito per il rispettivo Servizio AWS viene utilizzato per ultimo.

## Compatibilità con AWS gli SDK

I seguenti SDK supportano le funzionalità e le impostazioni descritte in questo argomento. Vengono annotate eventuali eccezioni parziali. Tutte le impostazioni delle proprietà del sistema JVM sono supportate solo da AWS SDK for Java and the. SDK AWS for Kotlin

SDK	Sì o	Note o ulteriori informazioni
<a href="#">AWS CLI v2</a>	Sì	
<a href="#">SDK per C++</a>	No	
<a href="#">SDK per Go V2 (1.x)</a>	Sì	
<a href="#">SDK per Go 1.x (V1)</a>	No	

SDK	Sì o	Note o ulteriori informazioni
<a href="#">SDK per Java 2.x</a>	No	
<a href="#">SDK per Java 1.x</a>	No	
<a href="#">SDK per 3.x JavaScript</a>	Sì	
<a href="#">SDK per 2.x JavaScript</a>	No	
<a href="#">SDK per Kotlin</a>	Sì	
<a href="#">SDK per .NET 3.x</a>	Sì	
<a href="#">SDK per PHP 3.x</a>	Sì	
<a href="#">SDK per Python (Boto3)</a>	Sì	
<a href="#">SDK per Ruby 3.x</a>	Sì	
<a href="#">SDK per Rust</a>	No	
<a href="#">Strumenti per PowerShell</a>	Sì	

## Identificatori per endpoint specifici del servizio

Per informazioni su come e dove utilizzare gli identificatori nella tabella seguente, vedere. [Endpoint specifici del servizio](#)

<b>serviceId</b>	Cl id at de se pe il fil cc Al co
AccessAnalyzer	ac AWS_ENDPOINT_URL_ACCESSANALYZER ly
Account	ac AWS_ENDPOINT_URL_ACCOUNT
ACM	ac AWS_ENDPOINT_URL_ACM
ACM PCA	ac AWS_ENDPOINT_URL_ACM_PCA
Alexa For Business	ac AWS_ENDPOINT_URL_ALEXA_FOR_BUSINESS _l
amp	ac AWS_ENDPOINT_URL_AMP
Amplify	ac AWS_ENDPOINT_URL_AMPLIFY
AmplifyBackend	ac AWS_ENDPOINT_URL_AMPLIFYBACKEND cl
AmplifyUIBuilder	ac AWS_ENDPOINT_URL_AMPLIFYUIBUILDER bt
API Gateway	ac AWS_ENDPOINT_URL_API_GATEWAY ay

<b>serviceId</b>	Cl id at de se pe il fil cc Al co	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
ApiGatewayManageme ntApi	a y n	AWS_ENDPOINT_URL_APIGATEWAYMANAGEMENTAPI
ApiGatewayV2	a y	AWS_ENDPOINT_URL_APIGATEWAYV2
AppConfig	a	AWS_ENDPOINT_URL_APPCONFIG
AppConfigData	a d	AWS_ENDPOINT_URL_APPCONFIGDATA
AppFabric	a	AWS_ENDPOINT_URL_APPFABRIC
Appflow	a	AWS_ENDPOINT_URL_APPFLOW
AppIntegrations	a a	AWS_ENDPOINT_URL_APPINTEGRATIONS
Application Auto Scaling	a o c	AWS_ENDPOINT_URL_APPLICATION_AUTO_SCALING

<b>serviceId</b>	Cl	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
Application Insights	ap	AWS_ENDPOINT_URL_APPLICATION_INSIGHTS
ApplicationCostProfiler	ap	AWS_ENDPOINT_URL_APPLICATIONCOSTPROFILER
App Mesh	ap	AWS_ENDPOINT_URL_APP_MESH
AppRunner	ap	AWS_ENDPOINT_URL_APPRUNNER
AppStream	ap	AWS_ENDPOINT_URL_APPSTREAM
AppSync	ap	AWS_ENDPOINT_URL_APPS SYNC
ARC Zonal Shift	a:	AWS_ENDPOINT_URL_ARC_ZONAL_SHIFT
Artifact	a:	AWS_ENDPOINT_URL_ARTIFACT
Athena	a:	AWS_ENDPOINT_URL_ATHENA
AuditManager	a:	AWS_ENDPOINT_URL_AUDITMANAGER

<b>serviceId</b>	Cl	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
	id at de se pe il fil cc Al co	
Auto Scaling	ai	AWS_ENDPOINT_URL_AUTO_SCALING
Auto Scaling Plans	ai	AWS_ENDPOINT_URL_AUTO_SCALING_PLANS
b2bi	b:	AWS_ENDPOINT_URL_B2BI
Backup	b:	AWS_ENDPOINT_URL_BACKUP
Backup Gateway	b:	AWS_ENDPOINT_URL_BACKUP_GATEWAY
BackupStorage	b:	AWS_ENDPOINT_URL_BACKUPSTORAGE
Batch	b:	AWS_ENDPOINT_URL_BATCH
BCM Data Exports	b:	AWS_ENDPOINT_URL_BCM_DATA_EXPORTS
Bedrock	b:	AWS_ENDPOINT_URL_BEDROCK
Bedrock Agent	b:	AWS_ENDPOINT_URL_BEDROCK_AGENT

<b>serviceId</b>	Cl	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
Bedrock Agent Runtime	b:	AWS_ENDPOINT_URL_BEDROCK_AGENT_RUNTIME
Bedrock Runtime	b:	AWS_ENDPOINT_URL_BEDROCK_RUNTIME
billingconductor	b:	AWS_ENDPOINT_URL_BILLINGCONDUCTOR
Braket	b:	AWS_ENDPOINT_URL_BRAKET
Budgets	b:	AWS_ENDPOINT_URL_BUDGETS
Cost Explorer	c:	AWS_ENDPOINT_URL_COST_EXPLORER
chatbot	c:	AWS_ENDPOINT_URL_CHATBOT
Chime	c:	AWS_ENDPOINT_URL_CHIME
Chime SDK Identity	c:	AWS_ENDPOINT_URL_CHIME_SDK_IDENTITY



<b>serviceId</b>	Cl Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b> id at de se pe il fil cc Al co
Chime SDK Media Pipelines	cl AWS_ENDPOINT_URL_CHIME_SDK_MEDIA_PIPELINES _f pe
Chime SDK Meetings	cl AWS_ENDPOINT_URL_CHIME_SDK_MEETINGS _f
Chime SDK Messaging	cl AWS_ENDPOINT_URL_CHIME_SDK_MESSAGING _f g
Chime SDK Voice	cl AWS_ENDPOINT_URL_CHIME_SDK_VOICE _f
CleanRooms	c: AWS_ENDPOINT_URL_CLEANROOMS s
CleanRoomsML	c: AWS_ENDPOINT_URL_CLEANROOMSML sr
Cloud9	c: AWS_ENDPOINT_URL_CLOUD9
CloudControl	c: AWS_ENDPOINT_URL_CLOUDCONTROL r

<b>serviceId</b>	Cl
	id at de se pe il fil cc Al co
CloudDirectory	c: AWS_ENDPOINT_URL_CLOUDDIRECTORY cl
CloudFormation	c: AWS_ENDPOINT_URL_CLOUDFORMATION at
CloudFront	c: AWS_ENDPOINT_URL_CLOUDFRONT t
CloudFront KeyValuesStore	c: AWS_ENDPOINT_URL_CLOUDFRONT_KEYVALUESTORE t_ es
CloudHSM	c: AWS_ENDPOINT_URL_CLOUDHSM
CloudHSM V2	c: AWS_ENDPOINT_URL_CLOUDHSM_V2 vi
CloudSearch	c: AWS_ENDPOINT_URL_CLOUDSEARCH cl
CloudSearch Domain	c: AWS_ENDPOINT_URL_CLOUDSEARCH_DOMAIN cl

<b>serviceId</b>	Cl Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
CloudTrail	cl AWS_ENDPOINT_URL_CLOUDTRAIL
CloudTrail Data	cl AWS_ENDPOINT_URL_CLOUDTRAIL_DATA
CloudWatch	cl AWS_ENDPOINT_URL_CLOUDWATCH
codeartifact	cl AWS_ENDPOINT_URL_CODEARTIFACT
CodeBuild	cl AWS_ENDPOINT_URL_CODEBUILD
CodeCatalyst	cl AWS_ENDPOINT_URL_CODECATALYST
CodeCommit	cl AWS_ENDPOINT_URL_CODECOMMIT
CodeDeploy	cl AWS_ENDPOINT_URL_CODEDEPLOY
CodeGuru Reviewer	cl AWS_ENDPOINT_URL_CODEGURU_REVIEWER

<b>serviceId</b>	Cl Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b> id at de se pe il fil cc Al co
CodeGuru Security	cc AWS_ENDPOINT_URL_CODEGURU_SECURITY Se
CodeGuruProfiler	cc AWS_ENDPOINT_URL_CODEGURUPROFILER Ic
CodePipeline	cc AWS_ENDPOINT_URL_CODEPIPELINE it
CodeStar	cc AWS_ENDPOINT_URL_CODESTAR
CodeStar connections	cc AWS_ENDPOINT_URL_CODESTAR_CONNECTIONS cc ns
codestar notificat ions	cc AWS_ENDPOINT_URL_CODESTAR_NOTIFICATIONS no ic
Cognito Identity	cc AWS_ENDPOINT_URL_COGNITO_IDENTITY de
Cognito Identity Provider	cc AWS_ENDPOINT_URL_COGNITO_IDENTITY_PROVIDER de Ic

<b>serviceId</b>	Cl id at de se pe il fil cc A c	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
Cognito Sync	cc y	AWS_ENDPOINT_URL_COGNITO_SYNC
Comprehend	cc d	AWS_ENDPOINT_URL_COMPREHEND
ComprehendMedical	cc dr	AWS_ENDPOINT_URL_COMPREHENDMEDICAL
Compute Optimizer	cc p	AWS_ENDPOINT_URL_COMPUTE_OPTIMIZER
Config Service	cc r	AWS_ENDPOINT_URL_CONFIG_SERVICE
Connect	cc	AWS_ENDPOINT_URL_CONNECT
Connect Contact Lens	cc o n	AWS_ENDPOINT_URL_CONNECT_CONTACT_LENS
ConnectCampaigns	cc m	AWS_ENDPOINT_URL_CONNECTCAMPAIGNS
ConnectCases	cc s	AWS_ENDPOINT_URL_CONNECTCASES

<b>serviceId</b>	Cl Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b> id at de se pe il fil cc Al co
ConnectParticipant	cc AWS_ENDPOINT_URL_CONNECTPARTICIPANT rt
ControlTower	cc AWS_ENDPOINT_URL_CONTROLTOWER we
Cost Optimization Hub	cc AWS_ENDPOINT_URL_COST_OPTIMIZATION_HUB m: ht
Cost and Usage Report Service	cc AWS_ENDPOINT_URL_COST_AND_USAGE_REPO us: RT_SERVICE o: cc
Customer Profiles	cc AWS_ENDPOINT_URL_CUSTOMER_PROFILES p:
DataBrew	d: AWS_ENDPOINT_URL_DATABREW
DataExchange	d: AWS_ENDPOINT_URL_DATAEXCHANGE n:
Data Pipeline	d: AWS_ENDPOINT_URL_DATA_PIPELINE l:

<b>serviceId</b>	Cl id at de se pe il fil cc Al co	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
DataSync	d:	AWS_ENDPOINT_URL_DATASYNC
DataZone	d:	AWS_ENDPOINT_URL_DATAZONE
DAX	d:	AWS_ENDPOINT_URL_DAX
Detective	d:	AWS_ENDPOINT_URL_DETECTIVE
Device Farm	d: r:	AWS_ENDPOINT_URL_DEVICE_FARM
DevOps Guru	d: r:	AWS_ENDPOINT_URL_DEVOPS_GURU
Direct Connect	d: n:	AWS_ENDPOINT_URL_DIRECT_CONNECT
Application Discovery Service	a: o: e: c:	AWS_ENDPOINT_URL_APPLICATION_DISCOVERY_SERVICE
DLM	d:	AWS_ENDPOINT_URL_DLM

<b>serviceId</b>	Cl	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
Database Migration Service	d:	AWS_ENDPOINT_URL_DATABASE_MIGRATION_SERVICE
DocDB	d:	AWS_ENDPOINT_URL_DOCDB
DocDB Elastic	d:	AWS_ENDPOINT_URL_DOCDB_ELASTIC
drs	d:	AWS_ENDPOINT_URL_DRS
Directory Service	d:	AWS_ENDPOINT_URL_DIRECTORY_SERVICE
DynamoDB	d:	AWS_ENDPOINT_URL_DYNAMODB
DynamoDB Streams	d:	AWS_ENDPOINT_URL_DYNAMODB_STREAMS
EBS	e:	AWS_ENDPOINT_URL_EBS
EC2	e:	AWS_ENDPOINT_URL_EC2
EC2 Instance Connect	e:	AWS_ENDPOINT_URL_EC2_INSTANCE_CONNECT



<b>serviceId</b>	Cl id at de se pe il fil cc Al co	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
ECR	e	AWS_ENDPOINT_URL_ECR
ECR PUBLIC	e c	AWS_ENDPOINT_URL_ECR_PUBLIC
ECS	e	AWS_ENDPOINT_URL_ECS
EFS	e	AWS_ENDPOINT_URL_EFS
EKS	e	AWS_ENDPOINT_URL_EKS
EKS Auth	e	AWS_ENDPOINT_URL_EKS_AUTH
Elastic Inference	e n	AWS_ENDPOINT_URL_ELASTIC_INFERENCE
ElastiCache	e h	AWS_ENDPOINT_URL_ELASTICACHE
Elastic Beanstalk	e e	AWS_ENDPOINT_URL_ELASTIC_BEANSTALK
Elastic Transcoder	e I	AWS_ENDPOINT_URL_ELASTIC_TRANSCODER

<b>serviceId</b>	Cl	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
Elastic Load Balancing	e:	AWS_ENDPOINT_URL_ELASTIC_LOAD_BALANCING
Elastic Load Balancing v2	e:	AWS_ENDPOINT_URL_ELASTIC_LOAD_BALANCING_V2
EMR	er	AWS_ENDPOINT_URL_EMR
EMR containers	er	AWS_ENDPOINT_URL_EMR_CONTAINERS
EMR Serverless	er	AWS_ENDPOINT_URL_EMR_SERVERLESS
EntityResolution	er	AWS_ENDPOINT_URL_ENTITYRESOLUTION
Elasticsearch Service	e:	AWS_ENDPOINT_URL_ELASTICSEARCH_SERVICE
EventBridge	e:	AWS_ENDPOINT_URL_EVENTBRIDGE

<b>serviceId</b>	Cl	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
Evidently	ev	AWS_ENDPOINT_URL_EVIDENTLY
finspace	f:	AWS_ENDPOINT_URL_FINSPEACE
finspace data	f:	AWS_ENDPOINT_URL_FINSPEACE_DATA
Firehose	f:	AWS_ENDPOINT_URL_FIREHOSE
fis	f:	AWS_ENDPOINT_URL_FIS
FMS	fr	AWS_ENDPOINT_URL_FMS
forecast	fo	AWS_ENDPOINT_URL_FORECAST
forecastquery	fo	AWS_ENDPOINT_URL_FORECASTQUERY
FraudDetector	f:	AWS_ENDPOINT_URL_FRAUDETECTOR
FreeTier	f:	AWS_ENDPOINT_URL_FREETIER
FSx	f:	AWS_ENDPOINT_URL_FSX
GameLift	g:	AWS_ENDPOINT_URL_GAMELIFT

<b>serviceId</b>	Cl id at de se pe il fil cc Al co	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
Glacier	g:	AWS_ENDPOINT_URL_GLACIER
Global Accelerator	g: c:	AWS_ENDPOINT_URL_GLOBAL_ACCELERATOR
Glue	g:	AWS_ENDPOINT_URL_GLUE
grafana	g:	AWS_ENDPOINT_URL_GRAFANA
Greengrass	g: s	AWS_ENDPOINT_URL_GREENGRASS
GreengrassV2	g: s\	AWS_ENDPOINT_URL_GREENGRASSV2
GroundStation	g: t:	AWS_ENDPOINT_URL_GROUNDSTATION
GuardDuty	g:	AWS_ENDPOINT_URL_GUARDDUTY
Health	h:	AWS_ENDPOINT_URL_HEALTH
HealthLake	h: e	AWS_ENDPOINT_URL_HEALTHLAKE

<b>serviceId</b>	Cl id at de se pe il fil cc A co	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
Honeycode	hc	AWS_ENDPOINT_URL_HONEYCODE
IAM	ia	AWS_ENDPOINT_URL_IAM
identitystore	ia to	AWS_ENDPOINT_URL_IDENTITYSTORE
imagebuilder	ia de	AWS_ENDPOINT_URL_IMAGEBUILDER
ImportExport	ia o:	AWS_ENDPOINT_URL_IMPORTEXPORT
Inspector	ia	AWS_ENDPOINT_URL_INSPECTOR
Inspector Scan	ia _:	AWS_ENDPOINT_URL_INSPECTOR_SCAN
Inspector2	ia 2	AWS_ENDPOINT_URL_INSPECTOR2
InternetMonitor	ia o:	AWS_ENDPOINT_URL_INTERNETMONITOR
IoT	ia	AWS_ENDPOINT_URL_IOT

<b>serviceId</b>	Cl Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b> id at de se pe il fil cc Al co
IoT Data Plane	io AWS_ENDPOINT_URL_IOT_DATA_PLANE p:
IoT Jobs Data Plane	io AWS_ENDPOINT_URL_IOT_JOBS_DATA_PLANE di e
IoT 1Click Devices Service	io AWS_ENDPOINT_URL_IOT_1CLICK_DEVICES_ k_ SERVICE _:
IoT 1Click Projects	io AWS_ENDPOINT_URL_IOT_1CLICK_PROJECTS k_ s
IoTAnalytics	io AWS_ENDPOINT_URL_IOTANALYTICS io
IotDeviceAdvisor	io AWS_ENDPOINT_URL_IOTDEVICEADVISOR a:
IoT Events	io AWS_ENDPOINT_URL_IOT_EVENTS s
IoT Events Data	io AWS_ENDPOINT_URL_IOT_EVENTS_DATA s:

<b>serviceId</b>	Cl	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
	id at de se pe il fil cc Al co	
IoT FleetHub	io	AWS_ENDPOINT_URL_IOTFLEETHUB
IoT FleetWise	io	AWS_ENDPOINT_URL_IOTFLEETWISE
IoT Secure Tunneling	io	AWS_ENDPOINT_URL_IOTSECURETUNNELING
IoT SiteWise	io	AWS_ENDPOINT_URL_IOTSITWISE
IoT ThingsGraph	io	AWS_ENDPOINT_URL_IOTTHINGSGRAPH
IoT TwinMaker	io	AWS_ENDPOINT_URL_IOTTWINMAKER
IoT Wireless	io	AWS_ENDPOINT_URL_IOT_WIRELESS
ivs	iv	AWS_ENDPOINT_URL_IVS
IVS RealTime	iv	AWS_ENDPOINT_URL_IVS_REALTIME

<b>serviceId</b>	Cl	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
ivschat	iv	AWS_ENDPOINT_URL_IVSCHAT
Kafka	k	AWS_ENDPOINT_URL_KAFKA
KafkaConnect	k	AWS_ENDPOINT_URL_KAFKACONNECT
kendra	k	AWS_ENDPOINT_URL_KENDRA
Kendra Ranking	k	AWS_ENDPOINT_URL_KENDRA_RANKING
Keyspaces	k	AWS_ENDPOINT_URL_KEYSPACES
Kinesis	k	AWS_ENDPOINT_URL_KINESIS
Kinesis Video Archived Media	k	AWS_ENDPOINT_URL_KINESIS_VIDEO_ARCHIVED_MEDIA
Kinesis Video Media	k	AWS_ENDPOINT_URL_KINESIS_VIDEO_MEDIA



<b>serviceId</b>	Cl	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
Kinesis Video Signaling	id at de se pe il fil cc Al co	k: AWS_ENDPOINT_URL_KINESIS_VIDEO_SIGNALING i: a:
Kinesis Video WebRTC Storage		k: AWS_ENDPOINT_URL_KINESIS_VIDEO_WEBRT i: C_STORAGE t: e:
Kinesis Analytics		k: AWS_ENDPOINT_URL_KINESIS_ANALYTICS n:
Kinesis Analytics V2		k: AWS_ENDPOINT_URL_KINESIS_ANALYTICS_V2 n: v:
Kinesis Video		k: AWS_ENDPOINT_URL_KINESIS_VIDEO i:
KMS		k: AWS_ENDPOINT_URL_KMS
LakeFormation		l: AWS_ENDPOINT_URL_LAKEFORMATION t:
Lambda		l: AWS_ENDPOINT_URL_LAMBDA

<b>serviceId</b>	Cl	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
Launch Wizard	l:	AWS_ENDPOINT_URL_LAUNCH_WIZARD
Lex Model Building Service	l:	AWS_ENDPOINT_URL_LEX_MODEL_BUILDING_SERVICE
Lex Runtime Service	l:	AWS_ENDPOINT_URL_LEX_RUNTIME_SERVICE
Lex Models V2	l:	AWS_ENDPOINT_URL_LEX_MODELS_V2
Lex Runtime V2	l:	AWS_ENDPOINT_URL_LEX_RUNTIME_V2
License Manager	l:	AWS_ENDPOINT_URL_LICENSE_MANAGER
License Manager Linux Subscriptions	l:	AWS_ENDPOINT_URL_LICENSE_MANAGER_LINUX_SUBSCRIPTIONS

<b>serviceId</b>	Cl id at de se pe il fil cc Al co
License Manager User Subscriptions	l: a: AWS_ENDPOINT_URL_LICENSE_MANAGER_USE R_SUBSCRIPTIONS e: i:
Lightsail	l: AWS_ENDPOINT_URL_LIGHTSAIL
Location	l: AWS_ENDPOINT_URL_LOCATION
CloudWatch Logs	c: h: AWS_ENDPOINT_URL_CLOUDWATCH_LOGS
CloudWatch Logs	c: h: AWS_ENDPOINT_URL_CLOUDWATCH_LOGS
LookoutEquipment	l: u: AWS_ENDPOINT_URL_LOOKOUTEQUIPMENT
LookoutMetrics	l: t: AWS_ENDPOINT_URL_LOOKOUTMETRICS
LookoutVision	l: s: AWS_ENDPOINT_URL_LOOKOUTVISION
m2	m: AWS_ENDPOINT_URL_M2

<b>serviceId</b>	Cl	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
Machine Learning	id at de se pe il fil cc Al co	m: AWS_ENDPOINT_URL_MACHINE_LEARNING e:
Macie2		m: AWS_ENDPOINT_URL_MACIE2
ManagedBlockchain		m: AWS_ENDPOINT_URL_MANAGEDBLOCKCHAIN o:
ManagedBlockchain Query		m: AWS_ENDPOINT_URL_MANAGEDBLOCKCHAIN_QUERY o: q:
Marketplace Agreement		m: AWS_ENDPOINT_URL_MARKETPLACE_AGREEMENT c: e:
Marketplace Catalog		m: AWS_ENDPOINT_URL_MARKETPLACE_CATALOG c: g
Marketplace Deployment		m: AWS_ENDPOINT_URL_MARKETPLACE_DEPLOYMENT c: m:

<b>serviceId</b>	Cl id at de se pe il fil cc Al co	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
Marketplace Entitlement Service	m c er v:	AWS_ENDPOINT_URL_MARKETPLACE_ENTITLEMENT_SERVICE
Marketplace Commerce Analytics	m c c i	AWS_ENDPOINT_URL_MARKETPLACE_COMMERCIAL_ANALYTICS
MediaConnect	m e	AWS_ENDPOINT_URL_MEDIACONNECT
MediaConvert	m e:	AWS_ENDPOINT_URL_MEDIACONVERT
MediaLive	m	AWS_ENDPOINT_URL_MEDIALIVE
MediaPackage	m a	AWS_ENDPOINT_URL_MEDIAPACKAGE
MediaPackage Vod	m a	AWS_ENDPOINT_URL_MEDIAPACKAGE_VOD

<b>serviceId</b>	Cl	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
MediaPackageV2	me	AWS_ENDPOINT_URL_MEDIAPACKAGEV2
MediaStore	me	AWS_ENDPOINT_URL_MEDIASTORE
MediaStore Data	me	AWS_ENDPOINT_URL_MEDIASTORE_DATA
MediaTailor	me	AWS_ENDPOINT_URL_MEDIATAILOR
Medical Imaging	me	AWS_ENDPOINT_URL_MEDICAL_IMAGING
MemoryDB	me	AWS_ENDPOINT_URL_MEMORYDB
Marketplace Metering	me	AWS_ENDPOINT_URL_MARKETPLACE_METERING
Migration Hub	me	AWS_ENDPOINT_URL_MIGRATION_HUB
mgn	me	AWS_ENDPOINT_URL_MGN

<b>serviceId</b>	Cl id at de se pe il fil cc Al co	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
Migration Hub Refactor Spaces	m: c: e:	AWS_ENDPOINT_URL_MIGRATION_HUB_REFAC _I TOR_SPACES
MigrationHub Config	m: h: g	AWS_ENDPOINT_URL_MIGRATIONHUB_CONFIG
MigrationHubOrches trator	m: h: t:	AWS_ENDPOINT_URL_MIGRATIONHUBORCHESTRATOR
MigrationHubStrategy	m: h: g:	AWS_ENDPOINT_URL_MIGRATIONHUBSTRATEGY
Mobile	m:	AWS_ENDPOINT_URL_MOBILE
mq	m:	AWS_ENDPOINT_URL_MQ
MTurk	m:	AWS_ENDPOINT_URL_MTURK
MWAA	m:	AWS_ENDPOINT_URL_MWAA
Neptune	n:	AWS_ENDPOINT_URL_NEPTUNE

<b>serviceId</b>	Cl	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
	id at de se pe il fil cc Al co	
Neptune Graph	ne	AWS_ENDPOINT_URL_NEPTUNE_GRAPH
neptunedata	ne	AWS_ENDPOINT_URL_NEPTUNEDATA
Network Firewall	ne	AWS_ENDPOINT_URL_NETWORK_FIREWALL
NetworkManager	ne	AWS_ENDPOINT_URL_NETWORKMANAGER
NetworkMonitor	ne	AWS_ENDPOINT_URL_NETWORKMONITOR
nimble	n:	AWS_ENDPOINT_URL_NIMBLE
OAM	o:	AWS_ENDPOINT_URL_OAM
Omics	or	AWS_ENDPOINT_URL_OMICS
OpenSearch	op	AWS_ENDPOINT_URL_OPENSEARCH
OpenSearchServerless	op	AWS_ENDPOINT_URL_OPENSEARCHSERVERLESS



<b>serviceId</b>	Cl	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
	id at de se pe il fil cc Al co	
OpsWorks	o:	AWS_ENDPOINT_URL_OPSWORKS
OpsWorksCM	o: m	AWS_ENDPOINT_URL_OPSWORKSCM
Organizations	o: ic	AWS_ENDPOINT_URL_ORGANIZATIONS
OSIS	o:	AWS_ENDPOINT_URL_OSIS
Outposts	o:	AWS_ENDPOINT_URL_OUTPOSTS
p8data	p:	AWS_ENDPOINT_URL_P8DATA
p8data	p:	AWS_ENDPOINT_URL_P8DATA
Panorama	p:	AWS_ENDPOINT_URL_PANORAMA
Payment Cryptography	p: r: h:	AWS_ENDPOINT_URL_PAYMENT_CRYPTOGRAPHY
Payment Cryptography Data	p: r: h:	AWS_ENDPOINT_URL_PAYMENT_CRYPTOGRAPHY_DATA

<b>serviceId</b>	Cl	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
Pca Connector Ad	pe ct	AWS_ENDPOINT_URL_PCA_CONNECTOR_AD
Personalize	pe ze	AWS_ENDPOINT_URL_PERSONALIZE
Personalize Events	pe ze	AWS_ENDPOINT_URL_PERSONALIZE_EVENTS
Personalize Runtime	pe ze e	AWS_ENDPOINT_URL_PERSONALIZE_RUNTIME
PI	p:	AWS_ENDPOINT_URL_PI
Pinpoint	p:	AWS_ENDPOINT_URL_PINPOINT
Pinpoint Email	p: er	AWS_ENDPOINT_URL_PINPOINT_EMAIL
Pinpoint SMS Voice	p: sr	AWS_ENDPOINT_URL_PINPOINT_SMS_VOICE

<b>serviceId</b>	Cl Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b> id at de se pe il fil cc Al co
Pinpoint SMS Voice V2	p: AWS_ENDPOINT_URL_PINPOINT_SMS_VOICE_V2 sr _\'
Pipes	p: AWS_ENDPOINT_URL_PIPES
Polly	p: AWS_ENDPOINT_URL_POLLY
Pricing	p: AWS_ENDPOINT_URL_PRICING
PrivateNetworks	p: AWS_ENDPOINT_URL_PRIVATENETWORKS tv
Proton	p: AWS_ENDPOINT_URL_PROTON
QBusiness	q: AWS_ENDPOINT_URL_QBUSINESS
QConnect	q: AWS_ENDPOINT_URL_QCONNECT
QLDB	q: AWS_ENDPOINT_URL_QLDB
QLDB Session	q: AWS_ENDPOINT_URL_QLDB_SESSION ic
QuickSight	q: AWS_ENDPOINT_URL_QUICKSIGHT t

<b>serviceId</b>	Cl id at de se pe il fil cc A C	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
RAM	ra	AWS_ENDPOINT_URL_RAM
rbin	rb	AWS_ENDPOINT_URL_RBIN
RDS	rd	AWS_ENDPOINT_URL_RDS
RDS Data	rd	AWS_ENDPOINT_URL_RDS_DATA
Redshift	rs	AWS_ENDPOINT_URL_REDSHIFT
Redshift Data	rd	AWS_ENDPOINT_URL_REDSHIFT_DATA
Redshift Serverless	rs	AWS_ENDPOINT_URL_REDSHIFT_SERVERLESS
Rekognition	rk	AWS_ENDPOINT_URL_REKOGNITION
repostspace	rp	AWS_ENDPOINT_URL_REPOSTSPACE
resiliencehub	rh	AWS_ENDPOINT_URL_RESILIENCEHUB

<b>serviceId</b>	Cl	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
Resource Explorer 2	re	AWS_ENDPOINT_URL_RESOURCE_EXPLORER_2
Resource Groups	rg	AWS_ENDPOINT_URL_RESOURCE_GROUPS
Resource Groups Tagging API	rg	AWS_ENDPOINT_URL_RESOURCE_GROUPS_TAGGING_API
RoboMaker	rm	AWS_ENDPOINT_URL_ROBOMAKER
RolesAnywhere	ra	AWS_ENDPOINT_URL_ROLESEANYWHERE
Route 53	r5	AWS_ENDPOINT_URL_ROUTE_53
Route53 Recovery Cluster	r5	AWS_ENDPOINT_URL_ROUTE53_RECOVERY_CLUSTER

<b>serviceId</b>	Cl	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
Route53 Recovery Control Config	id at de se pe il fil cc Al co	AWS_ENDPOINT_URL_ROUTE53_RECOVERY_CONTROL_CONFIG
Route53 Recovery Readiness	id e e:	AWS_ENDPOINT_URL_ROUTE53_RECOVERY_READINESS
Route 53 Domains	id d	AWS_ENDPOINT_URL_ROUTE_53_DOMAINS
Route53Resolver	id S	AWS_ENDPOINT_URL_ROUTE53RESOLVER
RUM	id	AWS_ENDPOINT_URL_RUM
S3	s:	AWS_ENDPOINT_URL_S3
S3 Control	s: l	AWS_ENDPOINT_URL_S3_CONTROL
S3Outposts	s: s	AWS_ENDPOINT_URL_S3OUTPOSTS

<b>serviceId</b>	Cl	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
SageMaker	s:	AWS_ENDPOINT_URL_SAGEMAKER
SageMaker A2I Runtime	s:	AWS_ENDPOINT_URL_SAGEMAKER_A2I_RUNTIME
Sagemaker Edge	s:	AWS_ENDPOINT_URL_SAGEMAKER_EDGE
SageMaker FeatureStore Runtime	s:	AWS_ENDPOINT_URL_SAGEMAKER_FEATURESTORE_RUNTIME
SageMaker Geospatial	s:	AWS_ENDPOINT_URL_SAGEMAKER_GEOSPATIAL
SageMaker Metrics	s:	AWS_ENDPOINT_URL_SAGEMAKER_METRICS
SageMaker Runtime	s:	AWS_ENDPOINT_URL_SAGEMAKER_RUNTIME

<b>serviceId</b>	Cl Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b> id at de se pe il fil cc Al co
savingsplans	s: AWS_ENDPOINT_URL_SAVINGSPLANS at
Scheduler	s: AWS_ENDPOINT_URL_SCHEDULER
schemas	s: AWS_ENDPOINT_URL_SCHEMAS
SimpleDB	s: AWS_ENDPOINT_URL_SIMPLEDB
Secrets Manager	s: AWS_ENDPOINT_URL_SECRETS_MANAGER at
SecurityHub	s: AWS_ENDPOINT_URL_SECURITYHUB ul
SecurityLake	s: AWS_ENDPOINT_URL_SECURITYLAKE al
ServerlessApplicat ionRepository	s: AWS_ENDPOINT_URL_SERVERLESSAPPLICATI s: ONREPOSITORY ic t
Service Quotas	s: AWS_ENDPOINT_URL_SERVICE_QUOTAS uc



<b>serviceId</b>	Cl	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
Service Catalog	s:	AWS_ENDPOINT_URL_SERVICE_CATALOG
Service Catalog AppRegistry	s:	AWS_ENDPOINT_URL_SERVICE_CATALOG_APP_REGISTRY
ServiceDiscovery	s:	AWS_ENDPOINT_URL_SERVICEDISCOVERY
SES	s:	AWS_ENDPOINT_URL_SES
SESV2	s:	AWS_ENDPOINT_URL_SESV2
Shield	s:	AWS_ENDPOINT_URL_SHIELD
signer	s:	AWS_ENDPOINT_URL_SIGNER
SimSpaceWeaver	s:	AWS_ENDPOINT_URL_SIMSPACEWEAVER
SMS	s:	AWS_ENDPOINT_URL_SMS
Snow Device Management	s:	AWS_ENDPOINT_URL_SNOW_DEVICE_MANAGEMENT

<b>serviceId</b>	Cl id at de se pe il fil cc Al co	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
Snowball	si	AWS_ENDPOINT_URL_SNOWBALL
SNS	si	AWS_ENDPOINT_URL_SNS
SQS	si	AWS_ENDPOINT_URL_SQS
SSM	si	AWS_ENDPOINT_URL_SSM
SSM Contacts	si ct	AWS_ENDPOINT_URL_SSM_CONTACTS
SSM Incidents	si er	AWS_ENDPOINT_URL_SSM_INCIDENTS
Ssm Sap	si	AWS_ENDPOINT_URL_SSM_SAP
SSO	si	AWS_ENDPOINT_URL_SSO
SSO Admin	si	AWS_ENDPOINT_URL_SSO_ADMIN
SSO OIDC	si	AWS_ENDPOINT_URL_SSO_OIDC
SFN	si	AWS_ENDPOINT_URL_SFN
Storage Gateway	si at	AWS_ENDPOINT_URL_STORAGE_GATEWAY

<b>serviceId</b>	Cl Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b> id at de se pe il fil cc Al co
STS	st AWS_ENDPOINT_URL_STS
SupplyChain	sc AWS_ENDPOINT_URL_SUPPLYCHAIN in
Support	su AWS_ENDPOINT_URL_SUPPORT
Support App	sa AWS_ENDPOINT_URL_SUPPORT_APP pi
SWF	sw AWS_ENDPOINT_URL_SWF
synthetics	sy AWS_ENDPOINT_URL_SYNTHETICS s
Textract	te AWS_ENDPOINT_URL_TEXTRACT
Timestream InfluxDB	ti AWS_ENDPOINT_URL_TIMESTREAM_INFLUXDB m_ b
Timestream Query	ti AWS_ENDPOINT_URL_TIMESTREAM_QUERY m_
Timestream Write	ti AWS_ENDPOINT_URL_TIMESTREAM_WRITE m_

<b>serviceId</b>	Cl Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b> id at de se pe il fil cc Al co
tnb	ti AWS_ENDPOINT_URL_TNB
Transcribe	t: AWS_ENDPOINT_URL_TRANSCRIBE e
Transfer	t: AWS_ENDPOINT_URL_TRANSFER
Translate	t: AWS_ENDPOINT_URL_TRANSLATE
TrustedAdvisor	t: AWS_ENDPOINT_URL_TRUSTEDADVISOR v:
VerifiedPermissions	ve AWS_ENDPOINT_URL_VERIFIEDPERMISSIONS e: s
Voice ID	vi AWS_ENDPOINT_URL_VOICE_ID
VPC Lattice	vj AWS_ENDPOINT_URL_VPC_LATTICE c:
WAF	w: AWS_ENDPOINT_URL_WAF
WAF Regional	w: AWS_ENDPOINT_URL_WAF_REGIONAL n:

<b>serviceId</b>	Cl	Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>
WAFV2	id at de se pe il fil cc Al co	w: AWS_ENDPOINT_URL_WAFV2
WellArchitected	w	AWS_ENDPOINT_URL_WELLARCHITECTED
Wisdom	w:	AWS_ENDPOINT_URL_WISDOM
WorkDocs	w	AWS_ENDPOINT_URL_WORKDOCS
WorkLink	w	AWS_ENDPOINT_URL_WORKLINK
WorkMail	w	AWS_ENDPOINT_URL_WORKMAIL
WorkMailMessageFlow	w	AWS_ENDPOINT_URL_WORKMAILMESSAGEFLOW
WorkSpaces	w	AWS_ENDPOINT_URL_WORKSPACES
WorkSpaces Thin Client	w	AWS_ENDPOINT_URL_WORKSPACES_THIN_CLIENT
WorkSpaces Web	w	AWS_ENDPOINT_URL_WORKSPACES_WEB

<b>serviceId</b>	Cl Variabile di ambiente <b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b> id at de se pe il fil cc A c
XRay	x: AWS_ENDPOINT_URL_XRAY

## Impostazioni predefinite di configurazione intelligente

Con la funzionalità Smart Configuration Defaults, AWS gli SDK possono fornire valori predefiniti e ottimizzati per altre impostazioni di configurazione.

Configura questa funzionalità utilizzando quanto segue:

**defaults\_mode**- impostazione dei AWS **config** file condivisi, **AWS\_DEFAULTS\_MODE**- variabile d'ambiente, **aws.defaultsMode**- Proprietà del sistema JVM: solo Java/Kotlin

Con questa impostazione, puoi scegliere una modalità che si allinea all'architettura dell'applicazione, che fornisce quindi valori predefiniti ottimizzati per l'applicazione. Se un'impostazione AWS SDK ha un valore impostato in modo esplicito, quel valore ha sempre la precedenza. Se un'impostazione AWS SDK non ha un valore impostato in modo esplicito e non `defaults_mode` è uguale a quella precedente, questa funzionalità può fornire valori predefiniti diversi per varie impostazioni ottimizzate per l'applicazione. Le impostazioni possono includere quanto segue: impostazioni di comunicazione HTTP, comportamento dei tentativi, impostazioni regionali degli endpoint del servizio e, potenzialmente, qualsiasi configurazione relativa all'SDK. I clienti che utilizzano questa funzionalità possono ottenere nuove impostazioni di configurazione predefinite personalizzate per scenari di utilizzo comuni. Se il tuo non `defaults_mode` è uguale a `legacy`, ti consigliamo di eseguire dei test dell'applicazione quando aggiorni l'SDK, poiché i valori predefiniti forniti potrebbero cambiare man mano che le best practice evolvono.

Valore predefinito: `legacy`

Nota: per impostazione predefinita, le nuove versioni principali degli SDK saranno impostate su `standard`

Valori validi:

- `legacy`— Fornisce impostazioni predefinite che variano in base all'SDK ed esistevano prima della creazione di `defaults_mode`
- `standard`— Fornisce i valori predefiniti più recenti consigliati che dovrebbero essere sicuri per l'esecuzione nella maggior parte degli scenari.
- `in-region`— Si basa sulla modalità `standard` e include un'ottimizzazione personalizzata per le applicazioni che effettuano chiamate Servizi AWS dall'interno della stessa Regione AWS.
- `cross-region`— Si basa sulla modalità `standard` e include un'ottimizzazione personalizzata per le applicazioni che effettuano chiamate Servizi AWS in una regione diversa.
- `mobile`— Si basa sulla modalità `standard` e include un'ottimizzazione personalizzata per le applicazioni mobili.
- `auto`— Si basa sulla modalità `standard` e include funzionalità sperimentali. L'SDK tenta di scoprire l'ambiente di runtime per determinare automaticamente le impostazioni appropriate. Il rilevamento automatico è basato sull'euristiche e non fornisce una precisione del 100%. Se non è possibile determinare l'ambiente di esecuzione, `standard` viene utilizzata la modalità. Il rilevamento automatico potrebbe interrogare [i metadati dell'istanza e i dati utente](#), il che potrebbe introdurre latenza. Se la latenza di avvio è fondamentale per la tua applicazione, ti consigliamo invece di sceglierne una esplicita. `defaults_mode`

Esempio di impostazione di questo valore nel `config` file:

```
[default]
defaults_mode = standard
```

I seguenti parametri potrebbero essere ottimizzati in base alla selezione di `defaults_mode`:

- `retryMode`— specifica in che modo l'SDK tenta di riprovare. Per informazioni, consulta [Comportamento di ripetizione](#).
- `stsRegionalEndpoints`— Specifica in che modo l'SDK determina l'endpoint del Servizio AWS che utilizza per comunicare con (). AWS Security Token Service AWS STS Per informazioni, consulta [AWS STS Endpoint regionalizzati](#).
- `s3UsEast1RegionalEndpoints`: specifica in che modo l'SDK determina l'endpoint del AWS servizio che utilizza per comunicare con Amazon S3 per la regione. `us-east-1`

- `connectTimeoutInMillis`— Dopo aver effettuato un tentativo di connessione iniziale su un socket, il periodo di tempo prima del timeout. Se il client non riceve il completamento dell'handshake di connessione, rinuncia e fallisce l'operazione.
- `tlsNegotiationTimeoutInMillis`— Il tempo massimo che un handshake TLS può impiegare dal momento in cui il messaggio CLIENT HELLO viene inviato al momento in cui il client e il server hanno completamente negoziato i codici e si sono scambiati le chiavi.

Il valore predefinito per ogni impostazione cambia a seconda di quella selezionata per l'applicazione. `defaults_mode` Questi valori sono attualmente impostati come segue (soggetti a modifiche):

Parametro	Modalità <b>standard</b>	Modalità <b>in-region</b>	Modalità <b>cross-region</b>	Modalità <b>mobile</b>
<code>retryMode</code>	standard	standard	standard	standard
<code>stsRegionalEndpoints</code>	regional	regional	regional	regional
<code>s3UsEast1RegionalEndpoints</code>	regional	regional	regional	regional
<code>connectTimeoutInMillis</code>	3100	1100	3100	30000
<code>tlsNegotiationTimeoutInMillis</code>	3100	1100	3100	30000

Ad esempio, `defaults_mode` se l'opzione selezionata fosse `standard`, il valore di `standard` verrebbe assegnato a `retry_mode` (dalle `retry_mode` opzioni valide) e il valore di `regional` verrebbe assegnato a `stsRegionalEndpoints` (dalle `stsRegionalEndpoints` opzioni valide).



## Compatibilità con gli AWS SDK

I seguenti SDK supportano le funzionalità e le impostazioni descritte in questo argomento. Vengono annotate eventuali eccezioni parziali. Tutte le impostazioni delle proprietà del sistema JVM sono supportate solo da AWS SDK for Java and the. SDK AWS for Kotlin

SDK	Supportato	Note o ulteriori informazioni
<a href="#">AWS CLI v2</a>	No	
<a href="#">SDK per C++</a>	Sì	Parametri non ottimizzati: stsRegionalEndpoints „s3UsEast1RegionalEndpoints . tlsNegotiationTimeoutInMillis
<a href="#">SDK per Go V2 (1.x)</a>	Sì	Parametri non ottimizzati: retryMode „ stsRegionalEndpoints s3UsEast1RegionalEndpoints
<a href="#">SDK per Go 1.x (V1)</a>	No	
<a href="#">SDK per Java 2.x</a>	Sì	Parametri non ottimizzati: stsRegionalEndpoints
<a href="#">SDK per Java 1.x</a>	No	
<a href="#">SDK per 3.x JavaScript</a>	Sì	Parametri non ottimizzati: stsRegionalEndpoints „s3UsEast1RegionalEndpoints . tlsNegotiationTimeoutInMillis connectTimeoutInMillis viene chiamatoconnectio nTimeout .

SDK	Supportato	Note o ulteriori informazioni
<a href="#">SDK per 2.x JavaScript</a>	No	
<a href="#">SDK per Kotlin</a>	No	
<a href="#">SDK per .NET 3.x</a>	Sì	Parametri non ottimizzati: <code>connectTimeoutInMilliseconds</code> , <code>tlsNegotiationTimeoutInMilliseconds</code>
<a href="#">SDK per PHP 3.x</a>	Sì	Parametri non ottimizzati: <code>tlsNegotiationTimeoutInMilliseconds</code>
<a href="#">SDK per Python (Boto3)</a>	Sì	Parametri non ottimizzati: <code>tlsNegotiationTimeoutInMilliseconds</code>
<a href="#">SDK per Ruby 3.x</a>	Sì	
<a href="#">SDK per Rust</a>	No	
<a href="#">Strumenti per PowerShell</a>	Sì	Parametri non ottimizzati: <code>connectTimeoutInMilliseconds</code> , <code>tlsNegotiationTimeoutInMilliseconds</code> .

# AWSLibrerie Common Runtime (CRT)

Le librerie AWS Common Runtime (CRT) sono una libreria di base degli SDK. Il CRT è una famiglia modulare di pacchetti indipendenti, scritta in C. Ogni pacchetto offre buone prestazioni e un ingombro minimo per le diverse funzionalità richieste. Queste funzionalità sono comuni e condivise tra tutti gli SDK e offrono un migliore riutilizzo, ottimizzazione e precisione del codice. I pacchetti sono:

- [awslabs/aws-c-auth](#): autenticazione AWS lato client (provider di credenziali standard e firma (sigv4))
- [awslabs/aws-c-cal](#): tipi primitivi crittografici, hash (MD5, SHA256, SHA256 HMAC), firmatari, AES
- [awslabs/aws-c-common](#): Strutture dati di base, tipi primitivi di threading/sincronizzazione, gestione del buffer, funzioni relative a stdlib
- [awslabs/aws-c-compression](#): Algoritmi di compressione (codifica/decodifica Huffman)
- [awslabs/aws-c-event-stream](#): elaborazione dei messaggi di flusso di eventi (headers, prelude, payload, crc/trailer), implementazione di chiamate di procedura remote (RPC) su flussi di eventi
- [awslabs/aws-c-http](#): implementazione in C99 delle specifiche HTTP/1.1 e HTTP/2
- [awslabs/aws-c-io](#): Socket (TCP, UDP), DNS, pipe, loop di eventi, canali, SSL/TLS
- [awslabs/aws-c-iot](#): implementazione C99 dell'integrazione dei servizi cloud AWS IoT con i dispositivi
- [awslabs/aws-c-mqtt](#): Protocollo di messaggistica standard e leggero per l'Internet of Things (IoT)
- [awslabs/aws-c-s3](#): implementazione della libreria C99 per la comunicazione con il servizio Amazon S3, progettata per massimizzare il throughput su istanze Amazon EC2 a larghezza di banda elevata
- [awslabs/aws-c-sdkutils](#): Una libreria di utilità per l'analisi e la gestione dei profili AWS
- [awslabs/aws-checksums](#): CRC32c e CRC32 multiplatforma con accelerazione hardware con riserva di implementazioni software efficienti
- [awslabs/aws-lc](#): libreria crittografica generica gestita dal team di AWS crittografia AWS e dai suoi clienti, basata sul codice del progetto Google BoringSSL e del progetto OpenSSL
- [awslabs/s2n](#): Implementazione C99 dei protocolli TLS/SSL, progettata per essere piccola e veloce con la sicurezza come priorità

Il CRT è disponibile tramite tutti gli SDK tranne Go.

## Dipendenze CRT

Le librerie CRT formano una rete complessa di relazioni e dipendenze. Conoscere queste relazioni è utile se è necessario creare il CRT direttamente dal codice sorgente. Tuttavia, la maggior parte degli utenti accede alla funzionalità CRT tramite l'SDK del linguaggio (come AWS SDK per C++ o SDK AWS per Java) o l'SDK del dispositivo IoT del linguaggio (come IoT SDK per C++ o IoT SDK AWS per Java). Nel diagramma seguente, la casella Language CRT Bindings si riferisce al pacchetto che include le librerie CRT per un SDK linguistico specifico. Questa è una raccolta di pacchetti del modulo `aws-crt-*`, dove `*` è un linguaggio SDK (come o). [aws-crt-cppaws-crt-java](#)

Di seguito è riportata un'illustrazione delle dipendenze gerarchiche delle librerie CRT.

# Manutenzione e supporto

Per una panoramica degli strumenti su cui è possibile sviluppare applicazioni AWS, consulta [Tools to Build on AWS](#). Per informazioni sul supporto, consulta il [AWS Knowledge Center](#).

I seguenti argomenti riguardano le politiche di manutenzione e supporto delle versioni per gli AWS SDK.

## Argomenti

- [AWSPolitica di manutenzione degli SDK e degli strumenti](#)
- [AWS Matrice di supporto delle versioni degli SDK e degli strumenti](#)
- [Toolkit IDE](#)

## AWSPolitica di manutenzione degli SDK e degli strumenti

### Panoramica

Questo documento delinea la politica di manutenzione per i kit e gli strumenti di sviluppo AWS software (SDK), inclusi gli SDK per dispositivi mobili e IoT, e le relative dipendenze sottostanti. AWS fornisce regolarmente agli AWS SDK e agli strumenti aggiornamenti che possono contenere supporto per AWS API nuove o aggiornate, nuove funzionalità, miglioramenti, correzioni di bug, patch di sicurezza o aggiornamenti della documentazione. Gli aggiornamenti possono anche riguardare le modifiche relative alle dipendenze, ai runtime delle lingue e ai sistemi operativi. AWS Le versioni SDK vengono pubblicate nei gestori di pacchetti (ad esempio Maven, NuGet PyPI) e sono disponibili come codice sorgente su. GitHub

Consigliamo agli utenti di rimanere up-to-date con le versioni SDK per tenersi aggiornati sulle funzionalità più recenti, sugli aggiornamenti di sicurezza e sulle dipendenze sottostanti. L'uso continuato di una versione SDK non supportata non è consigliato e viene eseguito a discrezione dell'utente.

### Controllo delle versioni

Le versioni di rilascio dell'AWSSDK sono in formato X.Y.Z dove X rappresenta la versione principale. L'aumento della versione principale di un SDK indica che questo SDK ha subito modifiche significative e sostanziali per supportare nuovi idiomi e modelli nel linguaggio. Le versioni principali vengono introdotte quando le interfacce pubbliche (ad esempio classi, metodi, tipi, ecc.),

i comportamenti o la semantica sono cambiati. Le applicazioni devono essere aggiornate per poter funzionare con la versione SDK più recente. È importante aggiornare le versioni principali con attenzione e in conformità con le linee guida per l'aggiornamento fornite da AWS.

## Ciclo di vita della versione principale dell'SDK

Il ciclo di vita delle principali versioni di SDK e Tools è costituito da 5 fasi, descritte di seguito.

- **Anteprima per sviluppatori (Fase 0):** durante questa fase, gli SDK non sono supportati, non devono essere utilizzati negli ambienti di produzione e sono pensati solo per l'accesso anticipato e il feedback. È possibile che le versioni future introducano modifiche sostanziali. Una volta AWS identificata una versione come prodotto stabile, può contrassegnarla come Release Candidate. Le Release Candidate sono pronte per la versione GA, a meno che non emergano bug significativi, e riceveranno un supporto completo AWS.
- **Disponibilità generale (GA) (Fase 1):** durante questa fase, gli SDK sono completamente supportati. AWS fornirà versioni SDK regolari che includono il supporto per nuovi servizi, aggiornamenti delle API per i servizi esistenti e correzioni di bug e sicurezza. Per Tools, AWS fornirà versioni regolari che includono nuovi aggiornamenti delle funzionalità e correzioni di bug. AWS supporterà la versione GA di un SDK per almeno 24 mesi.
- **Annuncio di manutenzione (Fase 2):** AWS pubblicherà un annuncio pubblico almeno 6 mesi prima che un SDK entri in modalità di manutenzione. Durante questo periodo, l'SDK continuerà a essere completamente supportato. In genere, la modalità di manutenzione viene annunciata contemporaneamente al passaggio della versione principale successiva a GA.
- **Manutenzione (Fase 3):** durante la modalità di manutenzione, AWS limita le versioni SDK per risolvere solo le correzioni di bug e i problemi di sicurezza critici. Un SDK non riceverà aggiornamenti delle API per servizi nuovi o esistenti né verrà aggiornato per supportare nuove regioni. La modalità di manutenzione ha una durata predefinita di 12 mesi, se non diversamente specificato.
- **Fine del supporto (Fase 4):** quando un SDK raggiunge la fine del supporto, non riceverà più aggiornamenti o versioni. Le versioni pubblicate in precedenza continueranno a essere disponibili tramite gestori di pacchetti pubblici e il codice rimarrà attivo. GitHub Il GitHub repository può essere archiviato. L'uso di un SDK raggiunto end-of-support viene effettuato a discrezione dell'utente. Consigliamo agli utenti di eseguire l'aggiornamento alla nuova versione principale.

Di seguito è riportata un'illustrazione visiva del ciclo di vita della versione principale dell'SDK. Tieni presente che le tempistiche riportate di seguito sono illustrative e non vincolanti.

## Ciclo di vita delle dipendenze

La maggior parte degli AWS SDK ha dipendenze sottostanti, come i runtime del linguaggio, i sistemi operativi o le librerie e i framework di terze parti. Queste dipendenze sono in genere legate alla comunità linguistica o al fornitore proprietario di quel particolare componente. Ogni comunità o fornitore pubblica la propria end-of-support pianificazione per il proprio prodotto.

I seguenti termini vengono utilizzati per classificare le dipendenze sottostanti di terze parti:

- **Sistema operativo (OS):** alcuni esempi includono Amazon Linux AMI, Amazon Linux 2, Windows 2008, Windows 2012, Windows 2016, ecc.
- **Language Runtime:** gli esempi includono Java 7, Java 8, Java 11, .NET Core, .NET Standard, .NET PCL, ecc.
- **Libreria/Framework di terze parti:** gli esempi includono OpenSSL, .NET Framework 4.5, Java EE, ecc.

La nostra politica prevede di continuare a supportare le dipendenze SDK per almeno 6 mesi dopo la fine del supporto per la dipendenza da parte della community o del fornitore. Questa politica, tuttavia, potrebbe variare a seconda della dipendenza specifica.

### Note

AWS si riserva il diritto di interrompere il supporto per una dipendenza sottostante senza aumentare la versione principale dell'SDK

## Metodi di comunicazione

Gli annunci di manutenzione vengono comunicati in diversi modi:

- Agli account interessati viene inviato un annuncio via e-mail che annuncia i nostri piani per terminare il supporto per la versione SDK specifica. L'e-mail illustrerà il percorso da seguire end-of-support, specificherà le tempistiche della campagna e fornirà indicazioni per l'aggiornamento.
- AWS La documentazione SDK, come la documentazione di riferimento sulle API, le guide per l'utente, le pagine di marketing dei prodotti SDK e i GitHub readme, viene aggiornata per indicare la tempistica della campagna e fornire indicazioni sull'aggiornamento delle applicazioni interessate.

- Viene pubblicato un post AWS sul blog che delinea il percorso e ribadisce le tempistiche della end-of-support campagna.
- Gli avvisi di obsolescenza vengono aggiunti agli SDK, delineando il percorso e il collegamento alla documentazione SDK. end-of-support

Per visualizzare l'elenco delle versioni principali disponibili di AWS SDK e strumenti e la relativa fase del ciclo di manutenzione, consulta. [the section called “Matrice di supporto delle versioni”](#)

## AWS Matrice di supporto delle versioni degli SDK e degli strumenti

La matrice seguente mostra l'elenco delle versioni principali del AWS Software Development Kit (SDK) disponibili e la loro posizione nel ciclo di vita della manutenzione con le relative tempistiche. Per informazioni dettagliate sul ciclo di vita delle versioni principali di AWS SDK e strumenti e sulle relative dipendenze sottostanti, consulta. [the section called “Politica di manutenzione”](#)

SDK	Versione principale	Fase attuale	Data di disponibilità generale	Note
<a href="#">AWS CLI</a>	1.x	Disponibilità generale	2/09/2013	
<a href="#">AWS CLI</a>	2.x	Disponibilità generale	10/2/2020	
<a href="#">SDK per C++</a>	1.x	Disponibilità generale	2/09/2015	
<a href="#">SDK per Go V2</a>	Versione 2 1.x	Disponibilità generale	19/01/2021	
<a href="#">SDK for Go</a>	1.x	Annuncio di manutenzione	19/11/2015	Vedi <a href="#">l'annuncio o per dettagli</a> e date
<a href="#">SDK per Java</a>	1.x	Annuncio di manutenzione	25/03/2010	Vedi <a href="#">l'annuncio o per dettagli</a> e date



SDK	Versione principale	Fase attuale	Data di disponibilità generale	Note
<a href="#">SDK per Java</a>	2.x	Disponibilità generale	20/11/2018	
<a href="#">SDK per JavaScript</a>	1.x	Fine del supporto	6/05/2013	
<a href="#">SDK per JavaScript</a>	2.x	Annuncio di manutenzione	19/06/2014	Vedi <a href="#">l'annuncio o per dettagli e date</a>
<a href="#">SDK per JavaScript</a>	3.x	Disponibilità generale	15/12/2020	
<a href="#">SDK per Kotlin</a>	1.x	Disponibilità generale	27/11/2023	
<a href="#">SDK per .NET</a>	1.x	Fine del supporto	11/2009	
<a href="#">SDK per .NET</a>	2.x	Fine del supporto	8/11/2013	
<a href="#">SDK per .NET</a>	3.x	Disponibilità generale	28/07/2015	
<a href="#">SDK per PHP</a>	2.x	Fine del supporto	02/11/2012	
<a href="#">SDK per PHP</a>	3.x	Disponibilità generale	27/05/2015	
<a href="#">SDK per Python (Boto2)</a>	1.x	Fine del supporto	13/07/2011	
<a href="#">SDK per Python (Boto3)</a>	1.x	Disponibilità generale	22/06/2015	

SDK	Versione principale	Fase attuale	Data di disponibilità generale	Note
<a href="#">SDK per Python (Botocore)</a>	1.x	Disponibilità generale	22/06/2015	
<a href="#">SDK per Ruby</a>	1.x	Fine del supporto	14/07/2011	
<a href="#">SDK per Ruby</a>	2.x	Fine del supporto	15/02/2015	
<a href="#">SDK per Ruby</a>	3.x	Disponibilità generale	29/08/2017	
<a href="#">SDK per Rust</a>	1.x	Disponibilità generale	27/11/2023	
<a href="#">SDK per Swift</a>	1.x	Anteprima per sviluppatori		
<a href="#">Strumenti per PowerShell</a>	2.x	Fine del supporto	8/11/2013	
<a href="#">Utensili per PowerShell</a>	3.x	Fine del supporto	29/07/2015	
<a href="#">Utensili per PowerShell</a>	4.x	Disponibilità generale	21/11/2019	

## Toolkit IDE

AWSI toolkit IDE (Integrated Development Environment) sono plugin ed estensioni che consentono l'accesso ai AWS servizi dell'IDE.

Per informazioni dettagliate su ciascuno dei Toolkit IDE, consulta queste guide per l'utente del Toolkit:

- [AWS Toolkit for Visual Studio](#)

- [AWS Toolkit for Visual Studio Code](#)
- [AWS Toolkit for JetBrains](#)

Le seguenti sezioni contengono informazioni di supporto, report di manutenzione e notifiche per AWS IDE Toolkit.

## Notifica di telemetria

AWSIDE Toolkit può raccogliere e archiviare dati di telemetria lato client per informare le decisioni relative alle future versioni del Toolkit. AWS I dati raccolti quantificano l'utilizzo del Toolkit. AWS

[Per ulteriori informazioni sui dati di telemetria raccolti in tutti i toolkit AWS IDE, consulta il documento CommonDefinitions.json nel repository Github.](#) `aws-toolkit-common`

Per informazioni dettagliate sui dati di telemetria raccolti da ciascuno dei Toolkit IDE, fai riferimento ai documenti di risorse nei seguenti repository Github di Toolkit: AWS AWS

- [AWS Toolkit for Visual Studio](#)
- [AWS Toolkit for Visual Studio Code](#)
- [AWS Toolkit for JetBrains](#)

Alcuni AWS servizi accessibili dai Toolkit possono raccogliere dati di telemetria lato client aggiuntivi, come Amazon. CodeWhisperer Per informazioni dettagliate sul tipo di dati raccolti CodeWhisperer o su come disattivare la telemetria lato client CodeWhisperer, consulta l'[argomento Condivisione dei dati](#) con AWS nella Amazon User Guide. CodeWhisperer

# Guida di riferimento alla cronologia dei documenti per AWS SDK e strumenti

La tabella seguente descrive importanti aggiunte e aggiornamenti alla Guida di riferimento agli AWS SDK e agli strumenti. Per ricevere notifiche sugli aggiornamenti della documentazione, puoi sottoscrivere il feed RSS.

Modifica	Descrizione	Data
<a href="#">Aggiornamenti delle impostazioni</a>	Aggiungere le impostazioni di configurazione del sistema JVM.	27 marzo 2024
<a href="#">Aggiornamenti della tabella di compatibilità</a>	Aggiornamenti alla compatibilità per il supporto SDK, aggiornamenti alle procedure IAM Identity Center.	20 febbraio 2024
<a href="#">Aggiornamento delle credenziali del contenitore.</a> <a href="#">Aggiornamento IMDS.</a>	Aggiunta del supporto per Amazon EKS. Aggiunta un'impostazione per disabilitare il fallback di IMDSv1.	29 dicembre 2023
<a href="#">Richiedi la compressione</a>	Aggiungere impostazioni per la funzionalità di compressione delle richieste.	27 dicembre 2023
<a href="#">Tabelle di compatibilità</a>	Tabelle di compatibilità per SDK e funzionalità degli strumenti aggiornate per includere SDK per Kotlin, SDK per Rust e. AWS Tools for PowerShell	10 dicembre 2023
<a href="#">Aggiornamenti di autenticazione</a>	Aggiornamenti ai metodi di autenticazione supportati per SDK e strumenti.	1 luglio 2023

<a href="#">Aggiornamenti delle best practice di IAM</a>	Guida aggiornata per l'allineamento alle best practice IAM. Per ulteriori informazioni, consulta <a href="#">Best practice per la sicurezza in IAM</a> .	27 febbraio 2023
<a href="#">Aggiornamenti SSO</a>	Aggiornamenti alle credenziali SSO per la nuova configurazione del token SSO.	19 novembre 2022
<a href="#">Aggiornamenti delle impostazioni</a>	Aggiornamenti alla tabella di supporto per la configurazione generale e per i punti di accesso multiregionali di Amazon S3.	17 novembre 2022
<a href="#">Aggiornamenti delle impostazioni</a>	Aggiornamenti alla chiarezza del client IMDS e delle credenziali IMDS. Aggiornamenti alle variabili di ambiente.	4 novembre 2022
<a href="#">Aggiornamento della pagina di benvenuto</a>	Annuncio di Amazon CodeWhisperer.	22 settembre 2022
<a href="#">Modifica del nome del servizio per Single Sign-On</a>	Aggiornamenti che riflettono il fatto che l' AWS SSO viene ora denominato. AWS IAM Identity Center	26 luglio 2022
<a href="#">Aggiornamento delle impostazioni</a>	Aggiornamenti minori ai dettagli del file di configurazione e alle impostazioni supportate.	15 giugno 2022
<a href="#">Aggiorna</a>	Aggiornamento massiccio di quasi tutte le parti di questa guida.	1 febbraio 2022

[Versione iniziale](#)

La prima versione di questa guida viene rilasciata al pubblico.

13 marzo 2020

# Glossario AWS

Per la terminologia AWS più recente, consultare il [glossario AWS](#) nella documentazione di riferimento per 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.