



Guia de referência

# AWS SDKs e ferramentas



# AWS SDKs e ferramentas: Guia de referência

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

As marcas comerciais e imagens comerciais da Amazon não podem ser usadas no contexto de nenhum produto ou serviço que não seja da Amazon, nem de qualquer maneira que possa gerar confusão entre os clientes ou que deprecie ou desprestige a Amazon. Todas as outras marcas comerciais que não são propriedade da Amazon pertencem aos respectivos proprietários, os quais podem ou não ser afiliados, estar conectados ou ser patrocinados pela Amazon.

# Table of Contents

AWS Guia de referência de SDKs e ferramentas .....	1
Recursos para desenvolvedores .....	2
Notificação de telemetria do kit de ferramentas .....	3
Configuração .....	4
Arquivos config e credentials compartilhados .....	5
Perfis .....	5
Formato do arquivo de configuração .....	7
Formato do arquivo de credenciais .....	10
Localização de arquivos compartilhados .....	11
Resolução do diretório inicial .....	11
Alterar a localização padrão desses arquivos .....	12
Variáveis de ambiente .....	13
Como definir variáveis de ambiente .....	13
Configuração de variável de ambiente sem servidor .....	15
Propriedades do sistema JVM .....	15
Como definir as propriedades do sistema JVM .....	16
Autenticação e acesso .....	18
ID do builder AWS .....	20
Autenticação do IAM Identity Center .....	20
Configure o acesso programático usando o Centro de Identidade do IAM .....	21
Entenda a autenticação do IAM Identity Center .....	24
IAM Roles Anywhere .....	28
Etapa 1: configurar IAM Roles Anywhere .....	28
Etapa 2: usar IAM Roles Anywhere .....	29
Assumir uma função .....	30
Assumir um perfil do IAM .....	31
Federar com identidade web ou OpenID Connect .....	32
AWS chaves de acesso .....	34
Use credenciais de curto prazo .....	34
Use credenciais de longo prazo .....	34
Credenciais de curto prazo .....	35
Credenciais de longo prazo .....	37
Perfis do IAM para instâncias do Amazon EC2 .....	40
Criar um perfil do IAM .....	40

Executar uma instância do Amazon EC2 e especificar seu perfil do IAM .....	41
Conectar à instância do EC2 .....	41
Executar a amostra de aplicativo na instância do EC2 .....	41
Referência de configurações .....	43
Criar clientes de serviço .....	43
Precedência de configurações .....	43
Lista de configurações de arquivo Config .....	45
Lista de configurações de arquivo Credentials .....	48
Lista de variáveis de ambiente .....	49
Lista de propriedades do sistema JVM .....	53
Provedores de credenciais padronizados .....	55
Cadeia de fornecedores de credenciais .....	56
AWS chaves de acesso .....	58
Assuma o provedor de perfil .....	61
Provedor de contêiner .....	67
Provedor do IAM Identity Center .....	71
Provedor de IMDS .....	77
Provedor de processo .....	82
Atributos padronizados .....	85
ID da aplicação .....	86
Metadados da instância do Amazon EC2 .....	88
Pontos de acesso Amazon S3 .....	91
Pontos de acesso de várias regiões do Amazon S3 .....	93
Região da AWS .....	95
AWS STS Endpoints regionalizados .....	98
Endpoints de pilha dupla e FIPS .....	101
Descoberta de endpoint .....	103
Configuração geral .....	105
Cliente de IMDS .....	108
Comportamento de repetição .....	111
Compactação de solicitações .....	117
Endpoints específicos de serviço .....	119
Padrões de configuração inteligentes .....	168
Comom runtime .....	173
Adicionar dependências .....	174
Política de manutenção .....	175

---

Visão geral .....	175
Versionamento .....	175
Ciclo de vida da versão principal do SDK .....	175
Ciclo de vida da dependência .....	176
Métodos de comunicação .....	177
Matriz de suporte a versões .....	178
Histórico do documento .....	181
Glossário do AWS .....	184
.....	clxxxv

# AWS Guia de referência de SDKs e ferramentas

Muitos SDKs e ferramentas compartilham algumas funcionalidades comuns, seja por meio de especificações de design compartilhadas ou por meio de uma biblioteca compartilhada.

Este guia inclui informações sobre:

- [Configuração](#)— Como usar os `credentials` arquivos `config` compartilhados ou variáveis de ambiente para configurar seus AWS SDKs e ferramentas.
- [Autenticação e acesso](#)— Estabeleça como seu código ou ferramenta se autentica AWS quando você desenvolve com Serviços da AWS.
- [Referência de configurações](#) – Referência para todas as configurações padronizadas disponíveis para autenticação e configuração.
- [Bibliotecas do Common Runtime \(CRT\) da AWS](#)— Visão geral das bibliotecas compartilhadas do AWS Common Runtime (CRT) que estão disponíveis para quase todos os SDKs.
- [AWS Política de manutenção de SDKs e ferramentas](#) abrange a política de manutenção e o controle de versões de kits de desenvolvimento de AWS software (SDKs) e ferramentas, incluindo SDKs para dispositivos móveis e Internet das Coisas (IoT) e suas dependências subjacentes.

Este guia de referência de AWS SDKs e ferramentas tem como objetivo ser uma base de informações aplicável a vários SDKs e ferramentas. O guia específico para o SDK ou ferramenta que você está usando deve ser usado além de qualquer informação apresentada aqui. A seguir estão o SDK e as ferramentas que têm seções relevantes do material neste guia:

Se você estiver usando:	As seções relevantes deste guia para você são:
<ul style="list-style-type: none"> <li>• Qualquer SDK ou ferramenta</li> </ul>	<a href="#">AWS Política de manutenção de SDKs e ferramentas</a>
<ul style="list-style-type: none"> <li>• <a href="#">AWS Cloud Development Kit (AWS CDK) Guia do desenvolvedor</a></li> <li>• <a href="#">AWS Serverless Application Model Guia do desenvolvedor</a></li> <li>• <a href="#">AWS Toolkit for Eclipse Guia do usuário</a></li> </ul>	<a href="#">Configuração</a> <a href="#">Autenticação e acesso</a> <a href="#">AWS Política de manutenção de SDKs e ferramentas</a>

Se você estiver usando:	As seções relevantes deste guia para você são:
<ul style="list-style-type: none"><li>• <a href="#">AWS Toolkit for JetBrains Guia do usuário</a></li><li>• <a href="#">AWS Toolkit for Visual Studio Guia do usuário</a></li><li>• <a href="#">AWS Toolkit for Visual Studio Code Guia do usuário</a></li></ul>	
<ul style="list-style-type: none"><li>• <a href="#">AWS Command Line Interface Guia do usuário</a></li><li>• <a href="#">AWS SDK for C++ Guia do desenvolvedor</a></li><li>• <a href="#">AWS SDK for Go Guia do desenvolvedor</a></li><li>• <a href="#">AWS SDK for Java Guia do desenvolvedor</a></li><li>• <a href="#">AWS SDK for JavaScript Guia do desenvolvedor</a></li><li>• <a href="#">AWS SDK para Kotlin</a></li><li>• <a href="#">AWS SDK for .NET Guia do desenvolvedor</a></li><li>• <a href="#">AWS SDK for PHP Guia do desenvolvedor</a></li><li>• <a href="#">AWS Introdução ao SDK for Python (Boto3)</a></li><li>• <a href="#">AWS SDK for Ruby Guia do desenvolvedor</a></li><li>• <a href="#">AWS SDK para Rust</a></li><li>• <a href="#">AWS SDK for Swift</a></li><li>• <a href="#">AWS Tools for Windows PowerShell Guia do usuário</a></li></ul>	<ul style="list-style-type: none"><li>• <a href="#">Configuração</a></li><li>• <a href="#">Autenticação e acesso</a></li><li>• <a href="#">Referência de configurações</a></li><li>• <a href="#">Bibliotecas do Common Runtime (CRT) da AWS</a></li><li>• <a href="#">AWS Política de manutenção de SDKs e ferramentas</a></li><li>• <a href="#">AWS Matriz de suporte de versões de SDKs e ferramentas</a></li></ul>

## Recursos para desenvolvedores

Para obter uma visão geral das ferramentas que podem ajudá-lo a desenvolver aplicativos AWS, consulte [Ferramentas para desenvolver AWS](#). Para obter informações sobre suporte, consulte a [Central de conhecimento AWS](#).

O Amazon Q Developer é um assistente conversacional generativo baseado em IA que pode ajudar você a entender, criar, estender e operar aplicativos. AWS Para acelerar sua construção AWS,

o modelo que impulsiona o Amazon Q é aprimorado com AWS conteúdo de alta qualidade para produzir respostas mais completas, acionáveis e referenciadas. Para obter mais informações, consulte [O que é o Amazon Q Developer?](#) no Amazon Q Developer User Guide.

## Notificação de telemetria do kit de ferramentas

AWS Os kits de ferramentas do Ambiente de Desenvolvimento Integrado (IDE) são plug-ins e extensões que permitem o acesso aos AWS serviços do seu IDE. Para obter informações detalhadas sobre cada um dos kits de ferramentas do IDE, consulte os Guias do usuário do kit de ferramentas na tabela anterior.

AWS Os kits de ferramentas do IDE podem coletar e armazenar dados de telemetria do lado do cliente para informar as decisões sobre versões futuras do kit de ferramentas. AWS Os dados coletados quantificam seu uso do Kit de AWS Ferramentas.

Para saber mais sobre os dados de telemetria coletados em todos os kits de ferramentas do AWS IDE, consulte o documento [commonDefinitions.json no repositório Github](#). `aws-toolkit-common`

Para obter informações detalhadas sobre os dados de telemetria coletados por cada um dos kits de ferramentas do AWS IDE, consulte os documentos de recursos nos seguintes repositórios do Github dos kits de AWS ferramentas:

- [AWS Toolkit for Visual Studio](#)
- [AWS Toolkit for Visual Studio Code](#)
- [AWS Toolkit for JetBrains](#)

Certos AWS serviços acessíveis nos AWS kits de ferramentas podem coletar dados adicionais de telemetria do lado do cliente. Para obter informações detalhadas sobre o tipo de dados coletados por cada AWS serviço individual, consulte o tópico de [AWS documentação](#) do serviço específico em que você está interessado.



# Configuração

Com AWS SDKs e outras ferramentas para AWS desenvolvedores, como o AWS Command Line Interface (AWS CLI), você pode interagir com as APIs AWS de serviço. Antes de tentar isso, no entanto, você deve configurar o SDK ou a ferramenta com as informações necessárias para realizar a operação solicitada.

Essas informações incluem os seguintes itens:

- Informações de credenciais que identificam quem está chamando a API. As credenciais são usadas para criptografar a solicitação para os AWS servidores. Usando essas informações, AWS confirma sua identidade e pode recuperar as políticas de permissões associadas a ela. Em seguida, ele pode determinar quais ações você tem permissão para realizar.
- Outros detalhes de configuração que você usa para informar ao SDK AWS CLI ou ao SDK como processar a solicitação, para onde enviar a solicitação (para qual endpoint de AWS serviço) e como interpretar ou exibir a resposta.

Cada SDK ou ferramenta oferece suporte a várias fontes que você pode usar para fornecer as informações de credenciais e de configuração necessárias. Algumas fontes são exclusivas do SDK ou da ferramenta, e você deve consultar a documentação dessa ferramenta ou do SDK para obter detalhes sobre como usar esse método.

No entanto, a maioria dos AWS SDKs e ferramentas oferece suporte a configurações comuns de duas fontes principais (além do código em si):

- Arquivos de [AWS configuração e credenciais compartilhados — Os arquivos](#) compartilhados `config` e `credentials` os arquivos são a forma mais comum de especificar a autenticação e a configuração em um AWS SDK ou ferramenta. Use esses arquivos para armazenar as configurações que suas ferramentas e aplicativos podem usar. As configurações nos arquivos `config` e `credentials` compartilhados estão associadas a um perfil específico. Com vários perfis, você pode criar configurações diferentes para aplicar em diferentes cenários. Ao usar uma AWS ferramenta para invocar um comando ou usar um SDK para invocar uma AWS API, você pode especificar qual perfil e, portanto, quais definições de configuração usar para essa ação. Um dos perfis é designado como o perfil `default` e é usado automaticamente quando você não especifica explicitamente um perfil a ser usado. As configurações que você pode armazenar nesses arquivos estão documentadas neste guia de referência.

- [Variáveis de ambiente](#) — Algumas das configurações podem ser armazenadas alternativamente nas variáveis de ambiente do seu sistema operacional. Embora você possa ter somente um conjunto de variáveis de ambiente em vigor por vez, elas são facilmente modificadas dinamicamente à medida que seu programa é executado e seus requisitos mudam.

Tópicos adicionais nesta seção

- [Arquivos config e credentials compartilhados](#)
- [Localização do compartilhado de arquivos config e credentials compartilhados](#)
- [Suporte a variáveis de ambiente](#)
- [Suporte às propriedades do sistema JVM](#)

## Arquivos **config** e **credentials** compartilhados

Os `credentials` arquivos compartilhados `AWS config` e contêm um conjunto de perfis. Um perfil é um conjunto de configurações, em pares chave-valor, usado pelo AWS Command Line Interface (AWS CLI), pelos AWS SDKs e por outras ferramentas. Os valores de configuração são anexados a um perfil para configurar algum aspecto do SDK/ferramenta quando esse perfil é usado. Esses arquivos são “compartilhados”, pois os valores afetam quaisquer aplicativos, processos ou SDKs no ambiente local de um usuário.

Tanto os arquivos `config` quanto `credentials` compartilhados são arquivos de texto simples que contêm somente caracteres ASCII (codificados em UTF-8). Eles assumem a forma do que geralmente é chamado de [arquivos INI](#).

### Perfis

As configurações nos arquivos `config` e `credentials` compartilhados estão associadas a um perfil específico. Vários perfis podem ser definidos no arquivo para criar configurações de configuração diferentes para serem aplicadas em diferentes ambientes de desenvolvimento.

O perfil `[default]` contém os valores que são usados por uma operação de SDK ou ferramenta se um perfil nomeado específico não for especificado. Você também pode criar perfis separados aos quais você pode referenciar explicitamente pelo nome. Cada perfil pode usar configurações e valores diferentes conforme necessário para seu aplicativo e cenário.

**Note**

[default] é simplesmente um perfil sem nome. Esse perfil é nomeado default porque é o perfil padrão usado pelo SDK se o usuário não especificar um perfil. Ele não fornece valores padrão herdados para outros perfis. Se você definir algo no [default] perfil e não o definir em um perfil nomeado, o valor não será definido quando você usar o perfil nomeado.

## Definir um perfil nomeado

O [default] perfil e vários perfis nomeados podem existir no mesmo arquivo. Use a configuração a seguir para selecionar quais configurações do perfil serão usadas pelo seu SDK ou ferramenta ao executar seu código. Os perfis também podem ser selecionados dentro do código ou por comando ao trabalhar com o AWS CLI

Configure essa funcionalidade definindo uma das seguintes opções:

### **AWS\_PROFILE**- variável de ambiente

Quando essa variável de ambiente é definida como um perfil nomeado ou “padrão”, todos os códigos e AWS CLI comandos do SDK usam as configurações desse perfil.

Exemplo de configuração de variáveis de ambiente para Linux/macOS por meio da linha de comando:

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

Exemplo do Windows de configuração de variáveis de ambiente por meio da linha de comando:

```
setx AWS_PROFILE "my_default_profile_name"
```

### **aws.profile**- Propriedade do sistema JVM

[Para o SDK para Kotlin na JVM e o SDK for Java 2.x, você pode definir a propriedade do sistema. `aws.profile`](#) Quando o SDK cria um cliente de serviço, ele usa as configurações no perfil nomeado, a menos que a configuração seja substituída no código. O SDK for Java 1.x não é compatível com essa propriedade do sistema.

## Formato do arquivo de configuração

O arquivo `config` é organizado em seções. Uma seção é um conjunto nomeado de configurações e continua até que outra linha de definição de seção seja encontrada.

O arquivo `config` é um arquivo de texto simples que usam o seguinte formato:

- Todas as entradas em uma seção assumem a forma geral de `setting-name=value`.
- As linhas podem ser comentadas iniciando-as com um caractere de hashtag (`#`).

### Tipos de seção

Uma definição de seção é uma linha que aplica um nome a uma coleção de configurações. As linhas de definição de seção começam e terminam com colchetes (`[ ]`). Dentro dos colchetes, há um identificador de tipo de seção e um nome personalizado para a seção. Você pode usar letras, números, hífens (`-`) e sublinhados (`_`), mas sem espaços.

Tipo de seção: **default**

Exemplo de linha de definição de seção: `[default]`

`[default]` é o único perfil que não exige o identificador da `profile` seção.

O exemplo a seguir mostra um arquivo `config` básico com um perfil `[default]`. Ele define a configuração [region](#). Todas as configurações que seguem essa linha, até que outra definição de seção seja encontrada, fazem parte desse perfil.

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

Tipo de seção: **profile**

Exemplo de linha de definição de seção: `[profile dev]`

A linha de definição da `profile` seção é um agrupamento de configuração nomeado que você pode aplicar a diferentes cenários de desenvolvimento. Para entender melhor os perfis nomeados, consulte a seção anterior sobre Perfis.

O exemplo a seguir mostra um config arquivo com uma linha de definição de profile seção e um perfil nomeado chamadofoo. Todas as configurações que seguem essa linha, até que outra definição de seção seja encontrada, fazem parte desse perfil nomeado.

```
[profile foo]
...settings...
```

Algumas configurações têm seu próprio grupo aninhado de subconfigurações, como a configuração e as subconfigurações de s3 no exemplo a seguir. Associe as subconfigurações ao grupo recuando-as com um ou mais espaços.

```
[profile test]
region = us-west-2
s3 =
    max_concurrent_requests=10
    max_queue_size=1000
```

### Tipo de seção: **sso-session**

Exemplo de linha de definição de seção: `[sso-session my-sso]`

A linha de definição da sso-session seção nomeia um grupo de configurações que você usa para configurar um perfil para resolver AWS as credenciais usando AWS IAM Identity Center. Para obter mais informações sobre como configurar a autenticação de login único, consulte [Autenticação do IAM Identity Center](#). Um perfil é vinculado a uma seção sso-session por um par de valores-chave em que sso-session é a chave e o nome da sua seção sso-session é o valor, como sso-session = <name-of-sso-session-section>.

O exemplo a seguir configura um perfil que obterá AWS credenciais de curto prazo para a função do IAM "SampleRole" na conta "111122223333" usando um token do "my-sso". A seção sso-session "my-sso" é referenciada na seção profile pelo nome usando a chave 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 de seção: **services**

Exemplo de linha de definição de seção: `[services dev]`

### Note

A seção `services` oferece suporte a personalizações de endpoints específicos do serviço e só está disponível em SDKs e ferramentas que incluem esse atributo. Para ver se esse atributo está disponível para seu SDK, consulte [Compatibilidade com AWS SDKs](#) para ver os endpoints específicos do serviço.

A linha de definição da `services` seção nomeia um grupo de configurações que configura endpoints personalizados para AWS service (Serviço da AWS) solicitações. Um perfil é vinculado a uma seção `services` por um par de valores-chave em que `services` é a chave e o nome da sua seção `services` é o valor, como `services = <name-of-services-section>`.

A `services` seção é ainda separada em subseções por `<SERVICE> =` linhas, onde `<SERVICE>` está a chave AWS service (Serviço da AWS) identificadora. O AWS service (Serviço da AWS) identificador é baseado no modelo de API, substituindo todos os espaços `serviceId` por sublinhados e colocando todas as letras em minúsculas. Para obter uma lista de todas as chaves de identificação de serviço a serem usadas na seção `services`, consulte [Identificadores para endpoints específicos de serviço](#). A chave de identificação de serviço é seguida por configurações aninhadas, cada uma em sua própria linha e recuada por dois espaços.

O exemplo a seguir usa uma definição `services` para configurar o endpoint a ser usado para solicitações feitas somente para o serviço Amazon DynamoDB . A seção `services "local-dynamodb"` é referenciada na seção `profile` pelo nome usando a chave `services`. A chave AWS service (Serviço da AWS) identificadora é `dynamodb`. A subseção de Amazon DynamoDB serviço começa na linha `dynamodb =` . Todas as linhas imediatamente seguintes que estejam recuadas são incluídas nessa subseção e se aplicam a esse serviço.

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

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



# Localização do compartilhado de arquivos **config** e **credentials** compartilhados

Os arquivos compartilhados AWS **config** e **credentials** são arquivos de texto simples que contêm informações de configuração para os AWS SDKs e as ferramentas. Os arquivos residem localmente em seu ambiente e são usados automaticamente pelo código do SDK ou pelos AWS CLI comandos que você executa nesse ambiente. Por exemplo, em seu próprio computador ou ao desenvolver em uma instância do Amazon Elastic Compute Cloud.

Quando o SDK ou a ferramenta são executados, eles verificam esses arquivos e carregam todas as configurações disponíveis. Se os arquivos ainda não existirem, um arquivo básico será criado automaticamente pelo SDK ou pela ferramenta.

Por padrão, os arquivos estão em uma pasta chamada `.aws` que é colocada na sua pasta home ou na pasta do usuário.

Sistema operacional	Local padrão e nome dos arquivos
Linux e macOS	<code>~/.aws/config</code> <code>~/.aws/credentials</code>
Windows	<code>%USERPROFILE%\.aws\config</code> <code>%USERPROFILE%\.aws\credentials</code>

## Resolução do diretório inicial

`~`só é usado para resolução de diretórios pessoais quando:

- Inicia o caminho
- É seguido imediatamente por `/` ou por um separador específico da plataforma. No Windows, `~/` e `~\` ambos são resolvidos para o diretório inicial.

Ao determinar o diretório inicial, as seguintes variáveis são verificadas:

- (Todas as plataformas) A variável de ambiente HOME



- (Plataformas Windows) A variável de ambiente USERPROFILE
- (Plataformas Windows) A concatenação de variáveis de HOMEDRIVE HOMEPAH ambiente ()  
\$HOMEDRIVE\$HOMEPAH
- (Opcional por SDK ou ferramenta) Um SDK ou função de resolução de caminho inicial específica do SDK ou da ferramenta

Quando possível, se o diretório inicial de um usuário for especificado no início do caminho (por exemplo, ~username/), ele será resolvido no diretório inicial do nome de usuário solicitado (por exemplo, /home/username/.aws/config).

## Alterar a localização padrão desses arquivos

Você pode usar qualquer uma das opções a seguir para substituir de onde esses arquivos são carregados pelo SDK ou pela ferramenta.

### Use variáveis de ambiente

As seguintes variáveis de ambiente podem ser definidas para alterar a localização ou o nome desses arquivos do valor padrão para um valor personalizado:

- Arquivo de variável de ambiente config: **AWS\_CONFIG\_FILE**
- Arquivo de variável de ambiente credentials: **AWS\_SHARED\_CREDENTIALS\_FILE**

### Linux/macOS

Você pode especificar um local alternativo executando os seguintes comandos de [exportação](#) no Linux ou no 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

Você pode especificar um local alternativo executando os seguintes comandos [setx](#) no 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
```

Para obter mais informações sobre como configurar seu sistema usando variáveis de ambiente, consulte [Suporte a variáveis de ambiente](#).

## Use as propriedades do sistema JVM

Para o SDK para Kotlin executado na JVM e para o SDK for Java 2.x, você pode definir as seguintes propriedades do sistema JVM para alterar a localização ou o nome desses arquivos do valor padrão para um valor personalizado:

- configpropriedade do sistema JVM do arquivo: **aws.configFile**
- Arquivo de variável de ambiente `credentials`: **aws.sharedCredentialsFile**

Para obter instruções sobre como definir as propriedades do sistema JVM, consulte [the section called “Como definir as propriedades do sistema JVM”](#). O SDK for Java 1.x não oferece suporte a essas propriedades do sistema.

## Suporte a variáveis de ambiente

Variáveis de ambiente fornecem outra maneira de especificar opções de configuração e credenciais e podem ser úteis para criação de scripts ou configuração temporária de um perfil nomeado como o padrão. Para ver a lista de variáveis de ambiente suportadas pela maioria dos SDKs, consulte [Lista de variáveis de ambiente](#).

### Precedência de opções

- Se você especificar uma configuração usando sua variável de ambiente, ela substituirá qualquer valor carregado de um perfil nos arquivos compartilhados AWS, `config` e `credentials`.
- Se você especificar uma configuração usando um parâmetro na linha de comando AWS CLI, ela substituirá qualquer valor da variável de ambiente correspondente ou de um perfil no arquivo de configuração.

## Como definir variáveis de ambiente

Os exemplos a seguir mostram como configurar variáveis de ambiente para o usuário padrão.

## 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
```

Configurar a variável de ambiente altera o valor usado até o final da sua sessão de shell ou até que você defina a variável como um valor diferente. Você pode tornar as variáveis persistentes em sessões futuras definindo-as no script de inicialização do shell.

## Windows Command Prompt

```
C:\> setx AWS_ACCESS_KEY_ID AKIAIOSFODNN7EXAMPLE
C:\> setx AWS_SECRET_ACCESS_KEY wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
C:\> setx
  AWS_SESSION_TOKEN AQoEXAMPLEH4aoAH0gNCAPy...truncated...zrkuWJ0gQs8IZZaIv2BXIa2R40lgk
C:\> setx AWS_REGION us-west-2
```

O uso de [set](#) para definir uma variável de ambiente altera o valor usado até o final da atual sessão de prompt de comando ou até que você defina a variável como um valor diferente. O uso de [setx](#) para definir uma variável de ambiente altera o valor usado na sessão atual de prompt de comando e todas as sessões de prompt de comando que você criar após a execução do comando. Não afeta outros shells de comando que já estejam em execução no momento em que você executar o comando.

## PowerShell

```
PS C:\> $Env:AWS_ACCESS_KEY_ID="AKIAIOSFODNN7EXAMPLE"
PS C:\> $Env:AWS_SECRET_ACCESS_KEY="wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY"
PS C:\>
  \> $Env:AWS_SESSION_TOKEN="AQoEXAMPLEH4aoAH0gNCAPy...truncated...zrkuWJ0gQs8IZZaIv2BXIa2R40lgk"
PS C:\> $Env:AWS_REGION="us-west-2"
```

Se você definir uma variável de ambiente no prompt do PowerShell conforme mostrado nos exemplos anteriores, ela salvará o valor somente pela duração da sessão atual. Para fazer com que a configuração da variável de ambiente seja persistente em todas as sessões do prompt de comando e do PowerShell, armazene-a usando o aplicativo System (Sistema) no Control Panel (Painel de controle). Como alternativa, você pode definir a variável para todas as futuras sessões do PowerShell adicionando-a ao seu perfil do PowerShell. Consulte a documentação do

[PowerShell](#) para obter mais informações sobre como armazenar variáveis de ambiente ou como persisti-las nas sessões.

## Configuração de variável de ambiente sem servidor

Se você usa uma arquitetura sem servidor para desenvolvimento, você tem outras opções para definir variáveis de ambiente. Dependendo do seu contêiner, você pode usar estratégias diferentes de execução de código nesses contêineres para ver e acessar as variáveis de ambiente, semelhantes a ambientes fora da nuvem.

Por exemplo, com AWS Lambda, você pode definir diretamente as variáveis de ambiente. Para obter mais detalhes, consulte [Usar AWS Lambda variáveis de ambiente](#) no AWS Lambda Guia do desenvolvedor.

No Serverless Framework, geralmente você pode definir variáveis de ambiente do SDK no arquivo `serverless.yml` sob a chave do provedor na configuração do ambiente. Para obter informações sobre o arquivo `serverless.yml`, consulte [Configurações gerais da função](#) na documentação do Serverless Framework.

Independentemente do mecanismo usado para definir variáveis de ambiente de contêiner, há algumas que são reservadas pelo contêiner, como aquelas documentadas para Lambda em [Defined runtime environment variables](#). Sempre consulte a documentação oficial do contêiner que você está usando para determinar como as variáveis de ambiente são tratadas e se há alguma restrição.

## Suporte às propriedades do sistema JVM

[As propriedades do sistema JVM](#) fornecem outra maneira de especificar opções de configuração e credenciais para SDKs executados na JVM, como o e o. AWS SDK for Java AWS SDK para Kotlin [Para obter uma lista das propriedades do sistema JVM suportadas pelos SDKs, consulte Referência de configurações.](#)

### Precedência de opções

- Se você especificar uma configuração usando sua propriedade de sistema JVM, ela substituirá qualquer valor encontrado nas variáveis de ambiente ou carregado de um perfil na AWS e nos arquivos compartilhados. `config credentials`
- Se você especificar uma configuração usando sua variável de ambiente, ela substituirá qualquer valor carregado de um perfil na AWS `config` e `credentials` nos arquivos compartilhados.

## Como definir as propriedades do sistema JVM

Você pode definir as propriedades do sistema JVM de várias maneiras.

### Na linha de comando

Defina as propriedades do sistema JVM na linha de comando ao invocar o `java` comando usando o `-D` switch. `-D` O comando a seguir configura Região da AWS globalmente para todos os clientes de serviço, a menos que você substitua explicitamente o valor no código.

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

Se você precisar definir várias propriedades do sistema JVM, especifique o `-D` switch várias vezes.

### Com uma variável de ambiente

Se você não conseguir acessar a linha de comando para invocar a JVM para executar seu aplicativo, poderá usar a variável de `JAVA_TOOL_OPTIONS` ambiente para configurar as opções da linha de comando. Essa abordagem é útil em situações como executar uma AWS Lambda função no Java Runtime ou executar código em uma JVM incorporada.

O exemplo a seguir configura Região da AWS globalmente para todos os clientes de serviço, a menos que você substitua explicitamente o valor no código.

Linux, macOS, or Unix

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

Configurar a variável de ambiente altera o valor usado até o final da sua sessão de shell ou até que você defina a variável como um valor diferente. Você pode tornar as variáveis persistentes em sessões futuras definindo-as no script de inicialização do shell.

Windows Command Prompt

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

O uso de [set](#) para definir uma variável de ambiente altera o valor usado até o final da atual sessão de prompt de comando ou até que você defina a variável como um valor diferente. O uso de [setx](#) para definir uma variável de ambiente altera o valor usado na sessão atual de prompt

de comando e todas as sessões de prompt de comando que você criar após a execução do comando. Não afeta outros shells de comando que já estejam em execução no momento em que você executar o comando.

## Em tempo de execução

Você também pode definir as propriedades do sistema JVM em tempo de execução no código usando o `System.setProperty` método, conforme mostrado no exemplo a seguir.

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

### Important

Defina todas as propriedades do sistema JVM antes de inicializar os clientes de serviço do SDK, caso contrário, os clientes de serviço poderão usar outros valores.

# Autenticação e acesso

Você precisa estabelecer como seu código deve ser autenticado com AWS ao desenvolver com Serviços da AWS. É possível configurar o acesso programático aos recursos AWS de maneiras diferentes, dependendo do ambiente e do acesso AWS disponível para você.

Opções de autenticação para código executado localmente (não em AWS)

- [Autenticação do IAM Identity Center](#) – Como prática recomendada de segurança, recomendamos o uso de AWS Organizations com o IAM Identity Center para gerenciar o acesso por todas as suas Contas da AWS. Você pode criar usuários em AWS IAM Identity Center, usar o Microsoft Active Directory, usar um provedor de identidade (IdP) SAML 2.0 ou federar seu IdP individualmente em Contas da AWS. Para verificar se sua Região é compatível com o IAM Identity Center, consulte [Endpoints e cotas AWS IAM Identity Center](#) no Referência geral da Amazon Web Services.
- [IAM Roles Anywhere](#) – Você pode usar o IAM Roles Anywhere para obter credenciais de segurança temporárias no IAM para workload, como servidores, - contêineres e aplicativos executados fora do AWS. Para usar o IAM Roles Anywhere, seu workload deve usar certificados X.509.
- [Assumir uma função](#) – Você pode assumir um perfil do IAM para acessar temporariamente recursos da AWS aos quais talvez não tivesse acesso de outra forma.
- [AWS chaves de acesso](#) – Outras opções que podem ser menos convenientes ou aumentar o risco de segurança de seus recursos AWS.

Opções de autenticação para código executado em um ambiente AWS

- [Usar perfis do IAM para instâncias do Amazon EC2](#) – Use o perfil do IAM para executar sua aplicação com segurança em uma instância Amazon EC2.
- Você pode interagir programaticamente com o AWS usando o IAM Identity Center das seguintes formas:
  - Use o [AWS CloudShell](#) para executar comandos AWS CLI a partir do console.
  - Use o [AWS Cloud9](#) para começar a programar no AWS usando um ambiente de desenvolvimento integrado (IDE) com recursos AWS.
  - Para tentar um espaço de colaboração baseado em nuvem para equipes de desenvolvimento de software, considere usar o [Amazon CodeCatalyst](#).

## Autenticação por meio de um provedor de identidades baseado na Web - Aplicações da web baseados em clientes ou móvel

Se você estiver criando aplicativos móveis ou aplicativos da web baseados em clientes que exigem acesso ao AWS, crie seu aplicativo para que ele solicite credenciais de segurança AWS temporárias de forma dinâmica usando a federação de identidades da web.

Com a federação de identidades da web, você não precisa criar código de login personalizado nem gerenciar suas próprias identidades de usuários. Em vez disso, os usuários do aplicativo podem fazer login usando um provedor de identidades (IdP) externo conhecido, como Login with Amazon, Facebook, Google ou qualquer outro IdP compatível com OpenID Connect (OIDC). Eles podem receber um token de autenticação e, em seguida, trocar esse token por credenciais de segurança temporárias na AWS que são mapeadas para um perfil do IAM com permissões para usar os recursos na sua Conta da AWS.

Para saber mais sobre como configurar isto para o seu SDK ou ferramenta, consulte [Federar com identidade web ou OpenID Connect](#).

Para aplicações móveis, recomendamos o uso do Amazon Cognito. O Amazon Cognito atua como um agente de identidades e realiza a maioria do trabalho de federação para você. Para obter mais informações, consulte [Uso do Amazon Cognito para aplicações móveis](#) no Guia do usuário do IAM.

### Mais informações sobre gerenciamento de acesso

O Guia do usuário do IAM tem as seguintes informações sobre o controle seguro do acesso aos recursos AWS:

- [Identidades do IAM \(usuários, grupos de usuários e perfis\)](#) – Entenda os fundamentos de identidades em AWS.
- [Melhores práticas de segurança no IAM](#) — Recomendações de segurança a serem seguidas ao desenvolver aplicativos AWS de acordo com o [modelo de responsabilidade compartilhada](#).

O Referência geral da Amazon Web Services tem noções básicas sobre o seguinte:

- [Entendendo e obtendo suas credenciais AWS](#) — Opções de chave de acesso e práticas de gerenciamento para acesso programático e de console.



# ID do builder AWS

Seu ID do builder AWS complementa qualquer Contas da AWS que você já possua ou queira criar. Enquanto uma Conta da AWS atua como um contêiner para os recursos da AWS que você cria e fornece um limite de segurança para esses recursos, seu ID do builder AWS representa você como um indivíduo. Você pode fazer login com a sua ID do builder AWS para acessar ferramentas e serviços para desenvolvedores, como Amazon CodeWhisperer e Amazon CodeCatalyst.

- [Fazer login no ID do builder AWS](#) no Guia do usuário do Início de Sessão da AWS — Saiba como criar e usar um ID do builder AWS e saiba o que o Builder ID fornece.
- [Autenticação com CodeWhisperer e AWS Toolkit - Builder ID](#) no Guia do usuário do CodeWhisperer - Saiba como o CodeWhisperer usa um ID do builder AWS.
- [Conceitos do CodeCatalyst — ID do builder AWS](#) no Guia do usuário do Amazon CodeCatalyst — Saiba como o CodeCatalyst usa um ID do builder AWS.

## Autenticação do IAM Identity Center

AWS IAM Identity Center é o método recomendado de fornecer AWS credenciais ao desenvolver em um serviço não AWS computacional. Por exemplo, isso seria algo como seu ambiente de desenvolvimento local. Se você estiver desenvolvendo em um AWS recurso, como o Amazon Elastic Compute Cloud (Amazon EC2) AWS Cloud9 ou, recomendamos que você obtenha credenciais desse serviço.

Neste tutorial, você estabelece o acesso ao IAM Identity Center e o configura para seu SDK ou ferramenta usando o portal de AWS acesso e o AWS CLI

- O portal de AWS acesso é o local da web em que você faz login manualmente no IAM Identity Center. O formato da URL é `d-xxxxxxxxxx.awsapps.com/start` ou `your_subdomain.awsapps.com/start`. Quando conectado ao portal de AWS acesso, você pode visualizar Contas da AWS as funções que foram configuradas para esse usuário. Esse procedimento usa o portal de AWS acesso para obter os valores de configuração necessários para o processo de autenticação do SDK/ferramenta.
- O AWS CLI é usado para configurar seu SDK ou ferramenta para usar a autenticação do IAM Identity Center para chamadas de API feitas pelo seu código. Esse processo único atualiza seu AWS config arquivo compartilhado, que é usado pelo SDK ou pela ferramenta quando você executa o código.

# Configure o acesso programático usando o Centro de Identidade do IAM

## Etapa 1: Estabelecer o acesso e selecionar o conjunto de permissões apropriado

Se você ainda não ativou o IAM Identity Center, consulte [Habilitar o IAM Identity Center](#) no Guia AWS IAM Identity Center do usuário.

Escolha um dos métodos a seguir para acessar suas AWS credenciais.

Não estabeleci acesso por meio do IAM Identity Center

1. Adicione um usuário e adicione permissões administrativas seguindo o procedimento [Configurar o acesso do usuário com o diretório padrão do IAM Identity Center](#) no Guia AWS IAM Identity Center do usuário.
2. O conjunto de `AdministratorAccess` permissões não deve ser usado para desenvolvimento regular. Em vez disso, recomendamos usar o conjunto de `PowerUserAccess` permissões predefinido, a menos que seu empregador tenha criado um conjunto de permissões personalizado para essa finalidade.

Siga o mesmo procedimento de [configuração do acesso do usuário com o procedimento de diretório padrão do IAM Identity Center](#) novamente, mas desta vez:

- Em vez de criar o *Admin team* grupo, crie um *Dev team* grupo e substitua-o posteriormente nas instruções.
- Você pode usar o usuário existente, mas o usuário deve ser adicionado ao novo *Dev team* grupo.
- Em vez de criar o *AdministratorAccess* conjunto de *PowerUserAccess* permissões, crie um conjunto de permissões e substitua-o posteriormente nas instruções.

Quando terminar, você deve ter o seguinte:

- Um `Dev team` grupo.
  - Um conjunto de `PowerUserAccess` permissões anexado ao `Dev team` grupo.
  - Seu usuário foi adicionado ao `Dev team` grupo.
3. Saia do portal e entre novamente para ver suas opções Contas da AWS e para `Administrator` ou `PowerUserAccess`. Selecione `PowerUserAccess` ao trabalhar com sua ferramenta/SDK.

Eu já tenho acesso AWS por meio de um provedor de identidade federado gerenciado pelo meu empregador (como Microsoft Entra ou Okta)

Faça login AWS por meio do portal do seu provedor de identidade. Se o seu administrador de nuvem concedeu permissões a você `PowerUserAccess` (desenvolvedor), você vê o Contas da AWS que você tem acesso e seu conjunto de permissões. Ao lado do nome do seu conjunto de permissões, você vê opções para acessar as contas manual ou programaticamente usando esse conjunto de permissões.

Implementações personalizadas podem resultar em experiências diferentes, como nomes de conjuntos de permissões diferentes. Se não tiver certeza sobre qual conjunto de permissões usar, entre em contato com a equipe de TI para obter ajuda.

Eu já tenho acesso a AWS através do portal de AWS acesso gerenciado pelo meu empregador

Faça login AWS por meio do portal de AWS acesso. Se o seu administrador de nuvem concedeu permissões a você `PowerUserAccess` (desenvolvedor), você vê o Contas da AWS que você tem acesso e seu conjunto de permissões. Ao lado do nome do seu conjunto de permissões, você vê opções para acessar as contas manual ou programaticamente usando esse conjunto de permissões.

Eu já tenho acesso AWS por meio de um provedor de identidade personalizado federado gerenciado pelo meu empregador

Entre em contato com a equipe de TI para obter ajuda.

## Etapa 2: configurar SDKs e ferramentas para usar o Centro de Identidade do IAM

1. Em sua máquina de desenvolvimento, instale o mais recente AWS CLI.
  - a. Consulte [Instalar ou atualizar a versão mais recente da AWS CLI](#) no Guia do usuário AWS Command Line Interface .
  - b. (Opcional) Para verificar se o AWS CLI está funcionando, abra um prompt de comando e execute o `aws --version` comando.
2. Faça login no portal de AWS acesso. Seu empregador pode fornecer esse URL ou você pode recebê-lo em um e-mail seguindo a Etapa 1: Estabelecer acesso. Caso contrário, encontre a URL do seu portal de AWS acesso no painel de controle de <https://console.aws.amazon.com/singlesignon/>.
  - a. No portal de AWS acesso, na guia Contas, selecione a conta individual a ser gerenciada. As funções do seu usuário são exibidas. Escolha Teclas de acesso para obter credenciais

- de linha de comando ou acesso programático para o conjunto de permissões apropriado. Use o conjunto de permissões `PowerUserAccess` predefinido ou qualquer conjunto de permissões que você ou seu empregador tenha criado para aplicar as permissões de privilégios mínimos para desenvolvimento.
- b. Na caixa de diálogo Obter credenciais, selecione MacOS e Linux ou Windows, dependendo do sistema operacional.
  - c. Selecione o método Credenciais IAM Identity Center para obter os valores `SSO Start URL` e `SSO Region` necessários para a próxima etapa.
3. No prompt de AWS CLI comando, execute o `aws configure sso` comando. Quando solicitado, insira os valores de configuração que você coletou na etapa anterior. Para obter detalhes sobre esse AWS CLI comando, consulte [Configurar seu perfil com o `aws configure sso` assistente](#).
- Para o nome do perfil CLI, recomendamos inserir o *padrão* ao começar. Para obter informações sobre como definir perfis não padrão (nomeados) e suas variáveis de ambiente associadas, consulte [Perfis](#).
4. (Opcional) No prompt de AWS CLI comando, confirme a identidade da sessão ativa executando o `aws sts get-caller-identity` comando. A resposta deve mostrar o conjunto de permissões do IAM Identity Center que você configurou.
5. Se você estiver usando um AWS SDK, crie um aplicativo para seu SDK em seu ambiente de desenvolvimento.
- a. Para alguns SDKs, pacotes adicionais, como `SSO` e `SSO0IDC` devem ser adicionados ao seu aplicativo antes que você possa usar a autenticação do IAM Identity Center. Para obter detalhes, consulte seu SDK específico.
  - b. Se você configurou anteriormente o acesso ao AWS, revise o `AWS credentials` arquivo compartilhado para verificar se há algum [AWS chaves de acesso](#). Você deve remover todas as credenciais estáticas antes que o SDK ou a ferramenta usem as credenciais do IAM Identity Center devido à precedência. [Cadeia de fornecedores de credenciais](#)

Para saber mais sobre como os SDKs e as ferramentas usam e atualizam as credenciais usando essa configuração, consulte [Entenda a autenticação do IAM Identity Center](#).

Dependendo da duração da sessão configurada, o seu acesso acabará expirando e o SDK ou ferramenta encontrarão um erro de autenticação. Para atualizar a sessão do portal de acesso novamente quando necessário, use o AWS CLI para executar o `aws sso login` comando.

Você pode estender a duração da sessão do portal de acesso do IAM Identity Center e a duração da sessão do conjunto de permissões. Isso aumenta a quantidade de tempo que você pode executar o código antes de precisar entrar manualmente novamente com o AWS CLI. Para obter mais informações, consulte os seguintes tópicos no Guia do usuário do AWS IAM Identity Center :

- Duração da sessão do IAM Identity Center — [Configure a duração das sessões do portal de AWS acesso de seus usuários](#)
- Permissão definir duração da sessão — [Definir duração da sessão](#)

Para obter detalhes sobre todas as configurações do provedor do IAM Identity Center para SDKs e ferramentas, consulte este guia [Provedor de credencial do IAM Identity Center](#).

## Entenda a autenticação do IAM Identity Center

### Termos relevantes do Centro de Identidade do IAM

Os termos a seguir ajudam você a entender o processo e a configuração por trás da AWS IAM Identity Center. A documentação das APIs do SDK AWS usa nomes diferentes do IAM Identity Center para alguns desses conceitos de autenticação. É útil conhecer os dois nomes.

A tabela a seguir mostra como os nomes alternativos se relacionam.

Nome do IAM Identity Center	Nome da API do SDK	Descrição
Identity Center	sso	Embora o Single Sign-On da AWS tenha sido renomeado , os namespaces da API sso manterão seu nome original para fins de compatibilidade com versões anteriores. Para obter mais informações, consulte <a href="#">Renomear o IAM Identity Center</a> no Guia do usuário AWS IAM Identity Center.
Console do IAM Identity Center		O console que você usa para configurar o single sign-on.

Nome do IAM Identity Center	Nome da API do SDK	Descrição
Console administrativo		
URL do portal de acesso da AWS		Um URL exclusivo para sua conta do IAM Identity Center, como <code>https://xxx.awsapps.com/start</code> . Você faz login neste portal usando suas credenciais de login do IAM Identity Center.
Sessão do portal de acesso ao IAM Identity Center	Sessão de autenticação	Fornece um token de acesso do portador ao chamador.
Sessão de definição de permissões		A sessão do IAM que o SDK usa internamente para fazer as chamadas de AWS service (Serviço da AWS). Em discussões informais, você pode ver isso incorretamente chamado de “sessão de funções”.
Credenciais do conjunto de permissões	Credenciais AWS credenciais sigv4	As credenciais que o SDK realmente usa para a maioria das chamadas de AWS service (Serviço da AWS) (especificamente, todas as chamadas AWS service (Serviço da AWS) sigv4). Em discussões informais, você pode ver isso incorretamente chamado de “credenciais de função”.

Nome do IAM Identity Center	Nome da API do SDK	Descrição
Provedor de credenciais do IAM Identity Center	Provedor de credenciais de SSO	Como você obtém as credenciais, como a classe ou o módulo que fornece a funcionalidade.

## Entenda a resolução de credenciais do SDK para Serviços da AWS

A API do IAM Identity Center troca as credenciais do token do portador por credenciais sigv4. A maioria dos Serviços da AWS são APIs sigv4, com algumas exceções, como Amazon CodeWhisperer e Amazon CodeCatalyst. A seguir, descrevemos o processo de resolução de credenciais para dar suporte à maioria das chamadas AWS service (Serviço da AWS) para o código do seu aplicativo por meio de AWS IAM Identity Center.

### Iniciar uma sessão do portal de acesso AWS

- Inicie o processo entrando na sessão com suas credenciais.
  - Use o comando `aws sso login` no AWS Command Line Interface (AWS CLI). Isso inicia uma nova sessão do IAM Identity Center se você ainda não tiver uma sessão ativa.
- Ao iniciar uma nova sessão, você recebe um token de atualização e um token de acesso do IAM Identity Center. O AWS CLI também atualiza um arquivo JSON de cache SSO com um novo token de acesso e token de atualização e o disponibiliza para uso por SDKs.
- Se você já tiver uma sessão ativa, o comando AWS CLI reutilizará a sessão existente e expirará sempre que a sessão existente expirar. Para saber como definir a duração de uma sessão do IAM Identity Center, consulte [Configurar a duração das sessões do portal de acesso AWS de seus usuários](#) no Guia do usuário AWS IAM Identity Center.
  - A duração máxima da sessão foi estendida para 90 dias para reduzir a necessidade de logins frequentes.

### Como o SDK obtém credenciais para chamadas AWS service (Serviço da AWS)

Os SDKs fornecem acesso para Serviços da AWS quando você instancia um objeto cliente por serviço. Quando o perfil selecionado do arquivo de `config` compartilhado da AWS é configurado para resolução de credenciais do IAM Identity Center, o IAM Identity Center é usado para resolver as credenciais do seu aplicativo.

- O [processo de resolução de credenciais](#) é concluído durante o runtime quando um cliente é criado.

Para recuperar as credenciais das APIs sigv4 usando o login único do IAM Identity Center, o SDK usa o token de acesso do IAM Identity Center para obter uma sessão do IAM. Essa sessão do IAM é chamada de sessão de conjunto de permissões e fornece acesso AWS ao SDK assumindo um perfil do IAM.

- A duração da sessão do conjunto de permissões é definida independentemente da duração da sessão do IAM Identity Center.
  - Para saber como definir a duração da sessão do conjunto de permissões, consulte [Definir a duração da sessão](#) no Guia do usuário AWS IAM Identity Center.
- Lembre-se de que as credenciais do conjunto de permissões também são chamadas de credenciais e credenciais AWS e credenciais sigv4 na maioria das documentações da API do SDK AWS.

As credenciais do conjunto de permissões são retornadas de uma chamada para [getRoleCredentials](#) da API IAM Identity Center para o SDK. O objeto cliente do SDK usa esse perfil do IAM assumido para fazer chamadas para o AWS service (Serviço da AWS), como pedir ao Amazon S3 que liste os buckets em sua conta. O objeto cliente pode continuar operando usando essas credenciais do conjunto de permissões até que a sessão do conjunto de permissões expire.

### Expiração e atualização da sessão

Ao usar o [Configuração do provedor de token do SSO](#), o token de acesso por hora obtido do IAM Identity Center é atualizado automaticamente usando o token de atualização.

- Se o token de acesso expirar quando o SDK tentar usá-lo, o SDK usará o token de atualização para tentar obter um novo token de acesso. O IAM Identity Center compara o token de atualização com a duração da sessão do portal de acesso do IAM Identity Center. Se o token de atualização não expirar, o IAM Identity Center responderá com outro token de acesso.
- Esse token de acesso pode ser usado para atualizar a sessão do conjunto de permissões de clientes existentes ou para resolver credenciais para novos clientes.

No entanto, se a sessão do portal de acesso do IAM Identity Center expirar, nenhum novo token de acesso será concedido. Portanto, a duração do conjunto de permissões não pode ser renovada. Ele



expirará (e o acesso será perdido) sempre que a duração da sessão definida em cache expirar para os clientes existentes.

Qualquer código que crie um novo cliente falhará na autenticação assim que a sessão do IAM Identity Center expirar. Isso ocorre porque as credenciais do conjunto de permissões não são armazenadas em cache. Seu código não conseguirá criar um novo cliente e concluir o processo de resolução de credenciais até que você tenha um token de acesso válido.

Para recapitular, quando o SDK precisa de novas credenciais de conjunto de permissões, ele primeiro verifica se há credenciais válidas existentes e as usa. Isso se aplica se as credenciais são para um novo cliente ou para um cliente existente com credenciais expiradas. Se as credenciais não forem encontradas ou não forem válidas, o SDK chama a API do IAM Identity Center para obter novas credenciais. Para chamar a API, ela precisa do token de acesso. Se o token de acesso expirar, o SDK usará o token de atualização para tentar obter um novo token de acesso a partir do serço IAM Identity Center. Esse token é concedido se sua sessão do portal de acesso ao IAM Identity Center não tiver expirado.

## IAM Roles Anywhere

Você pode usar o IAM Roles Anywhere para obter credenciais de segurança temporárias no IAM para workloads, como servidores, contêineres e aplicativos executados fora do AWS. Para usar o IAM Roles Anywhere, seu workload deve usar certificados X.509. Seu administrador de nuvem deve fornecer o certificado e a chave privada necessários para configurar o IAM Roles Anywhere como seu provedor de credenciais.

### Etapa 1: configurar IAM Roles Anywhere

O IAM Roles Anywhere fornece uma maneira de obter credenciais temporárias para um workload ou processo executado fora do AWS. Uma âncora de confiança é estabelecida com a autoridade de certificação para obter credenciais temporárias para o perfil do IAM associado. A função define as permissões que seu workload terá quando seu código for autenticado com o IAM Roles Anywhere.

Para ver as etapas para configurar a âncora de confiança, o perfil do IAM e o perfil do IAM Roles Anywhere, consulte [Como criar uma âncora de confiança e AWS Identity and Access Management perfil em Roles Anywhere](#) no Guia do usuário do IAM Roles Anywhere.

**Note**

Um perfil no Guia do usuário do IAM Roles Anywhere se refere a um conceito exclusivo no serviço IAM Roles Anywhere. Não está relacionado aos perfis no arquivo do config da AWS compartilhado.

## Etapa 2: usar IAM Roles Anywhere

Para obter credenciais de segurança temporárias do IAM Roles Anywhere, use a ferramenta de assistente de credenciais fornecida pelo IAM Roles Anywhere. A ferramenta de credenciais implementa o processo de assinatura do IAM Roles Anywhere.

Para obter instruções sobre como baixar a ferramenta de assistente de credenciais, consulte [Obter credenciais de segurança temporárias do Roles Anywhere AWS Identity and Access Management](#) no Guia do usuário do IAM Roles Anywhere.

Para usar credenciais de segurança temporárias do IAM Roles Anywhere com SDKs AWS e o AWS CLI, você pode definir a configuração `credential_process` no arquivo config da AWS compartilhado. Os SDKs AWS CLI oferecem suporte a um provedor de credenciais de processo que usa `credential_process` para autenticar. O seguinte mostra a estrutura geral a definir `credential_process`.

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

O comando `credential-process` da ferramenta auxiliar retorna credenciais temporárias em um formato JSON padrão compatível com a configuração `credential_process`. Observe que o nome do comando contém um hífen, mas o nome da configuração contém um sublinhado. O comando requer os seguintes parâmetros:

- `private-key` – O caminho para a chave privada que assinou a solicitação.
- `certificate` – O caminho para o certificado.
- `role-arn` – O ARN da função para a qual obter credenciais temporárias.
- `profile-arn` – O ARN do perfil que fornece um mapeamento para a função especificada.
- `trust-anchor-arn` – O ARN da âncora de confiança usada para autenticar.

Seu administrador de nuvem deve fornecer o certificado e uma chave privada. Todos os três valores de ARN podem ser copiados do AWS Management Console. O exemplo a seguir mostra um arquivo config compartilhado que configura a recuperação de credenciais temporárias da ferramenta auxiliar.

```
[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
```

Para parâmetros opcionais e detalhes adicionais da ferramenta auxiliar, consulte [Assistente de credenciais do IAM Roles Anywhere](#) no GitHub.

Para obter detalhes sobre a própria configuração do SDK e o provedor de credenciais do processo, consulte [Provedor de credenciais de processo](#) neste guia.

## Assumir uma função

Assumir um perfil envolve o uso de um conjunto de credenciais temporárias de segurança para acessar AWS recursos aos quais você talvez não tenha acesso de outra forma. Essas credenciais de segurança temporárias consistem em um ID de chave de acesso, uma chave de acesso secreta e um token de segurança. Para saber mais sobre as solicitações de API AWS Security Token Service (AWS STS), consulte [Ações](#) na Referência da API do AWS Security Token Service.

Para configurar seu SDK ou ferramenta para assumir um perfil, você deve primeiro criar ou identificar um perfil específico a ser assumido. [Os perfis do IAM são identificados exclusivamente por um perfil do nome do recurso da Amazon \(ARN\)](#). Os perfis estabelecem as relações de confiança com uma outra entidade. A entidade confiada que usa o perfil pode ser uma AWS service (Serviço da AWS), outra Conta da AWS, um provedor de identidade da web ou OIDC, ou uma federação SAML. Para obter mais informações sobre perfis do IAM, consulte [Perfis do IAM](#) no Guia do usuário do IAM.

Depois que perfil do IAM for identificado, se você tiver a confiança desse perfil, poderá configurar seu SDK ou ferramenta para usar as permissões concedidas pelo perfil. Para fazer isso, escolha [Assumir um perfil do IAM](#) ou [Federar com identidade web ou OpenID Connect](#).

## Assumir um perfil do IAM

Ao assumir um perfil, AWS STS retorna um conjunto de credenciais temporárias de segurança. Essas credenciais são provenientes de outro perfil ou da instância ou contêiner em que seu código está sendo executado. Outros exemplos de assumir um perfil incluem gerenciar várias contas Contas da AWS do Amazon EC2, usar AWS CodeCommit em Contas da AWS ou acessar outra conta a partir de AWS CodeBuild.

### Etapa 1: Configurar um perfil do IAM

Para configurar seu SDK ou ferramenta para assumir um perfil, você deve primeiro criar ou identificar um perfil específico a ser assumido. [Os perfis do IAM são identificados exclusivamente usando um ARN](#) de perfil. Os perfis estabelecem relações de confiança com outra entidade, normalmente dentro da sua conta ou para acesso entre contas. Para saber mais, consulte [Criação de perfis do IAM](#) no Guia do usuário do IAM.

### Etapa 2: Configurar o SDK ou a ferramenta

Configure o SDK ou a ferramenta para obter credenciais de `credential_source` ou `source_profile`.

Use `credential_source` para obter credenciais de um contêiner do Amazon ECS, de uma instância do Amazon EC2 ou de variáveis de ambiente.

Use `source_profile` para obter credenciais de outro perfil. `source_profile` também suporta o encadeamento de perfis, que são hierarquias de perfis em que um perfil assumido é então usado para assumir outro perfil.

Quando isso é especificado em um perfil, o SDK ou ferramenta faz automaticamente a chamada AWS STS [AssumeRole](#) API correspondente para você. Para recuperar e usar credenciais temporárias assumindo um perfil, especifique os seguintes valores de configuração no arquivo compartilhado AWS `config`. Para obter mais detalhes sobre cada uma dessas configurações, consulte a seção [Assuma as configurações do provedor de credenciais do perfil](#).

- `role_arn`- A partir do perfil do IAM que você criou na Etapa 1
- Configure um `source_profile` ou `credential_source`
- (Opcional) `duration_seconds`
- (Opcional) `external_id`

- (Opcional) `mfa_serial`
- (Opcional) `role_session_name`

Os exemplos a seguir mostram a configuração de ambas as opções de perfis assumidos em um arquivo compartilhado `config`:

```
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
```

Para obter mais detalhes sobre todas as configurações do provedor de credenciais para assumir o perfil, consulte este guia [Assuma o perfil de provedor de credenciais](#).

## Federar com identidade web ou OpenID Connect

Ao criar aplicativos móveis ou aplicativos web baseados em clientes que exigem acesso ao AWS, AWS STS retorna um conjunto de credenciais de segurança temporárias para usuários federados que são autenticados por meio de um provedor de identidades público (IdP). Exemplos de provedores de identidades públicos incluem Login with Amazon, Facebook, Google e qualquer provedor de identidades compatível com OpenID Connect (OIDC). Com este método, seus usuários não precisam de identidades próprias AWS ou do IAM.

Se você estiver usando o Amazon Elastic Kubernetes Service, este atributo oferece a capacidade de especificar diferentes perfis do IAM para cada um dos seus contêineres. O Kubernetes fornece a capacidade de distribuir tokens OIDC para seus contêineres, que são usados por este provedor de credenciais para obter credenciais temporárias. Para obter mais informações sobre essa configuração do Amazon EKS, consulte [Perfis do IAM para contas de serviço](#) no Guia do usuário do Amazon EKS. No entanto, como uma opção mais simples, recomendamos que você use o [Amazon EKS Pod Identities](#) se seu [SDK for compatível](#).

### Etapa 1: Configurar um provedor de identidades e um perfil do IAM

Para configurar uma federação com um IdP externo, crie um provedor de identidades do IAM para informar AWS sobre o IdP externo e sua configuração. Isso estabelece confiança entre seu Conta da AWS e o IdP externo. Antes de configurar o SDK para usar o token de identidade da web para

autenticação, você deve primeiro configurar o provedor de identidade (IdP) e o perfil do IAM usado para acessá-lo. Para configurá-los, consulte [Criar um perfil para identidades da Web ou federação do OpenID Connect \(console\)](#) no Guia do usuário do IAM.

## Etapa 2: Configurar o SDK ou a ferramenta

Configure o SDK ou a ferramenta para usar um token de identidade da web do AWS STS para autenticação.

Quando isso é especificado em um perfil, o SDK ou ferramenta faz automaticamente a chamada AWS STS [AssumeRoleWithWebIdentity](#) API correspondente para você. Para recuperar e usar credenciais temporárias usando a federação de identidades da web, é possível especificar os valores de configuração a seguir em um arquivo compartilhado AWS config. Para obter mais detalhes sobre cada uma dessas configurações, consulte a seção [Assuma as configurações do provedor de credenciais do perfil](#).

- `role_arn`- A partir do perfil do IAM que você criou na Etapa 1
- `web_identity_token_file`- Do IdP externo
- (Opcional) `duration_seconds`
- (Opcional) `role_session_name`

Veja a seguir um exemplo de uma configuração de arquivo config compartilhado para assumir um perfil com a identidade da web:

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

### Note

Para aplicações móveis, recomendamos o uso do Amazon Cognito. O Amazon Cognito atua como um agente de identidades e realiza a maioria do trabalho de federação para você. No entanto, o provedor de identidade do Amazon Cognito não está incluído nas bibliotecas principais de SDKs e ferramentas, como outros provedores de identidade. Para acessar a API do Amazon Cognito, inclua o cliente do serviço Amazon Cognito na compilação ou nas bibliotecas do seu SDK ou ferramenta. Para uso com AWS SDKs, consulte [exemplos de código](#) no Guia do desenvolvedor do Amazon Cognito.

Para obter mais detalhes sobre todas as configurações do provedor de credenciais para assumir o perfil, consulte este guia [Assuma o perfil de provedor de credenciais](#).

## AWS chaves de acesso

### Use credenciais de curto prazo

Recomendamos configurar o seu SDK ou ferramenta para usar [Autenticação do IAM Identity Center](#) para usar as opções de duração de sessão estendida.

No entanto, para configurar diretamente as credenciais temporárias do SDK ou da ferramenta, consulte [Autenticar usando credenciais de curto prazo](#).

### Use credenciais de longo prazo

#### Warning

Para evitar riscos de segurança, não use usuários do IAM para autenticação ao desenvolver software com propósito específico ou trabalhar com dados reais. Em vez disso, use federação com um provedor de identidade, como [AWS IAM Identity Center](#).

## Gerencie o acesso em Contas da AWS

Como prática recomendada de segurança, recomendamos o uso AWS Organizations com o IAM Identity Center para gerenciar o acesso em todos os seus Contas da AWS. Para obter mais informações, consulte [Práticas recomendadas de segurança no IAM](#) no Guia do usuário do IAM.

Você pode criar usuários no IAM Identity Center, usar o Microsoft Active Directory, usar um provedor de identidade (IdP) SAML 2.0 ou federar seu IdP individualmente para. Contas da AWS Usando uma dessas abordagens, você pode fornecer uma experiência de login único para seus usuários. Você também pode aplicar a autenticação multifator (MFA) e usar credenciais temporárias para acesso. Conta da AWS Isso difere de um usuário do IAM, que é uma credencial de longo prazo que pode ser compartilhada e que pode aumentar o risco de segurança de seus recursos AWS .

### Crie usuários do IAM somente para ambientes de sandbox

Se você é novato AWS, pode criar um usuário de teste do IAM e usá-lo para executar tutoriais e explorar o que AWS tem a oferecer. Não há problema em usar esse tipo de credencial quando você está aprendendo, mas recomendamos que você evite usá-la fora de um ambiente sandbox.

Para os seguintes casos de uso, pode fazer sentido começar com os usuários do IAM em AWS:

- Comece a usar seu AWS SDK ou ferramenta e explore Serviços da AWS em um ambiente sandbox.
- Executar scripts agendados, trabalhos e outros processos automatizados que não oferecem suporte a um processo de login assistido por humanos como parte de seu aprendizado.

Se você estiver usando usuários do IAM fora desses casos de uso, faça a transição para o IAM Identity Center ou federe seu provedor de identidade o mais rápido Contas da AWS possível. Para obter mais informações, consulte [Federação de identidades em AWS](#).

## Garanta chaves de acesso para usuários do IAM

Você deve alternar chaves de acesso de usuário do IAM regularmente. Siga as orientações em [Alternando chaves de acesso](#) no Guia do usuário do IAM. Se você acredita que compartilhou acidentalmente suas chaves de acesso de usuário do IAM, alterne suas chaves de acesso.

As chaves de acesso do usuário do IAM devem ser armazenadas no `AWS credentials` arquivo compartilhado na máquina local. Não armazene as chaves de acesso do usuário do IAM em seu código. Não inclua arquivos de configuração que contenham suas chaves de acesso de usuário do IAM em nenhum software de gerenciamento de código-fonte. Ferramentas externas, como o projeto de código aberto [git-secrets](#), podem ajudá-lo a evitar o envio inadvertido de informações confidenciais em um repositório Git. Para obter mais informações, consulte [Identidades IAM \(usuários, grupos e funções\)](#) no Guia Usuário do IAM.

Para configurar um usuário do IAM para começar, consulte [Autenticar com credenciais de longo prazo](#).

## Autenticar usando credenciais de curto prazo

Recomendamos configurar seu SDK ou ferramenta para usar [Autenticação do IAM Identity Center](#) com opções de duração de sessão estendida. No entanto, você pode copiar e usar credenciais temporárias que estão disponíveis no portal de acesso da AWS. As novas credenciais precisarão ser copiadas quando essas expirarem. É possível usar as credenciais temporárias em um perfil ou usá-las como valores para propriedades do sistema e variáveis de ambiente.





Depois que as credenciais temporárias expirarem, repita as etapas de 4 a 7.

## Autenticar com credenciais de longo prazo

### Warning

Para evitar riscos de segurança, não use usuários do IAM para autenticação ao desenvolver software com propósito específico ou trabalhar com dados reais. Em vez disso, use federação com um provedor de identidade, como [AWS IAM Identity Center](#).

Se você usa um usuário do IAM para executar seu código, o SDK ou a ferramenta em seu ambiente de desenvolvimento é autenticado usando credenciais de usuário do IAM de longo prazo no arquivo compartilhado. `AWS credentials` Analise o tópico [Melhores práticas de segurança no IAM](#) e faça a transição para o IAM Identity Center ou outras credenciais temporárias assim que possível.

## Avisos e orientações importantes para credenciais

### Avisos para credenciais

- NÃO use as credenciais de raiz da sua conta para acessar os recursos da AWS . Estas credenciais fornecem acesso ilimitado à conta e são difíceis de revogar.
- NÃO coloque chaves de acesso literais ou informações de credenciais nos comandos de seus aplicativos. Se colocar, criará um risco de exposição acidental das credenciais se, por exemplo, fizer upload do projeto em um repositório público.
- NÃO inclua arquivos que contenham credenciais em sua área de projeto.
- Esteja ciente de que todas as credenciais armazenadas no `AWS credentials` arquivo compartilhado são armazenadas em texto simples.

### Orientação adicional para gerenciar credenciais com segurança

Para uma discussão geral sobre como gerenciar AWS credenciais com segurança, consulte [Melhores práticas para gerenciar chaves de AWS acesso](#) no [Referência geral da AWS](#) Além dessa discussão, considere o seguinte:

- Use [perfis do IAM para tarefas](#) do Amazon Elastic Container Service (Amazon ECS).
- Use [perfis do IAM](#) para aplicações em execução nas instâncias do Amazon EC2.

## Pré-requisitos: Crie uma conta AWS

Para usar um usuário do IAM para acessar AWS serviços, você precisa de uma AWS conta e AWS credenciais.

### 1. Crie uma conta.

Para criar uma AWS conta, consulte [Primeiros passos: você é um AWS usuário iniciante?](#) no Guia AWS Account Management de referência.

### 2. Crie um usuário administrativo.

Evite usar a conta de usuário raiz (a conta inicial criada) para acessar serviços e o console de gerenciamento. Em vez disso, crie uma conta de usuário administrativo, conforme explicado em [Criar um usuário administrativo](#) no Guia do usuário do IAM.

Depois de criar a conta de usuário administrativo e registrar os detalhes de login, saia da conta de usuário raiz e faça login novamente usando a conta administrativa.

Nenhuma dessas contas é apropriada para desenvolvimento AWS ou execução de aplicativos AWS. Como prática recomendada, você precisa criar usuários, conjuntos de permissões ou perfis de serviço que sejam apropriados para essas tarefas. Para obter mais informações, consulte [Aplicar permissões de privilégio mínimo](#), no Guia do usuário do IAM.

## Etapa 1: criar o usuário do IAM

- Crie o usuário do IAM seguindo o procedimento de [Criação de usuários do IAM \(console\)](#) no Guia do usuário do IAM. Ao criar seu usuário do IAM:
  - Recomendamos que você selecione Fornecer acesso ao usuário ao AWS Management Console. Isso permite que você visualize informações Serviços da AWS relacionadas ao código que você está executando em um ambiente visual, como a verificação de registros de AWS CloudTrail diagnóstico ou o upload de arquivos para o Amazon Simple Storage Service, o que é útil ao depurar seu código.
  - Em Definir permissões - Opções de permissão, selecione Anexar políticas diretamente para saber como você deseja atribuir permissões a esse usuário.
    - A maioria dos tutoriais de “Conceitos básicos” do SDK usa o serviço Amazon S3 como exemplo. Para fornecer à aplicação acesso total ao Amazon S3, selecione a política AmazonS3FullAccess para anexar a esse usuário.

- Você pode ignorar as etapas opcionais desse procedimento em relação à definição de limites de permissão ou tags.

## Etapa 2: obter as chaves de acesso

1. No painel de navegação do console do IAM, selecione Usuários e escolha o usuário **User name** que você criou anteriormente.
2. Na página do usuário, selecione a página Credenciais de segurança. Depois, em Chaves de acesso, selecione Criar chave de acesso.
3. Para Criar chave de acesso: etapa 1, escolha interface de linha de comandos (CLI) ou Código local. Ambas as opções geram o mesmo tipo de chave para usar com os SDKs AWS CLI e os SDKs.
4. Em Criar chave de acesso: etapa 2, insira uma tag opcional e selecione Próximo.
5. Em Criar chave de acesso: etapa 3, selecione Baixar arquivo .csv para salvar um arquivo .csv com a chave de acesso e a chave de acesso secreta do usuário do IAM. Você precisará dessas informações posteriormente.

### Warning

Use medidas de segurança apropriadas para manter essas credenciais seguras.

6. Selecione Concluído.

## Etapa 3: atualizar o arquivo **credentials** compartilhado

1. Crie ou abra o arquivo AWS **credentials** compartilhado. Esse arquivo é `~/.aws/credentials` em sistemas Linux e macOS e `%USERPROFILE%\aws\credentials` no Windows. Para obter mais informações, consulte [Arquivos de credenciais de local](#).
2. Adicione o texto a seguir ao arquivo **credentials** compartilhado. Substitua o valor de ID de exemplo e o valor de chave de exemplo pelos valores no arquivo .csv que você baixou anteriormente.

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

### 3. Salve o arquivo.

O arquivo `credentials` compartilhado é a forma mais comum de armazenar credenciais. Eles também podem ser definidos como variáveis de ambiente, consulte [AWS chaves de acesso](#) para ver os nomes das variáveis de ambiente. Essa é uma forma de começar, mas recomendamos que você faça a transição para o IAM Identity Center ou outras credenciais temporárias o mais rápido possível. Depois de deixar de usar credenciais de longo prazo, lembre-se de excluir essas credenciais do arquivo `credentials` compartilhado.

## Usar perfis do IAM para instâncias do Amazon EC2

Este exemplo aborda a configuração de uma AWS Identity and Access Management função com acesso ao Amazon S3 para uso em seu aplicativo implantado em uma instância do Amazon EC2.

Para uma instância do Amazon Elastic Compute Cloud, crie um perfil do IAM e, em seguida, dê acesso a esse perfil à sua instância do Amazon EC2. Para obter mais informações, consulte [Funções do IAM para o Amazon EC2](#) no Guia do usuário do Amazon EC2 [ou Funções do IAM para o Amazon EC2 no Guia do usuário do Amazon](#) EC2.

### Criar um perfil do IAM

Crie um perfil do IAM que conceda acesso somente leitura ao Amazon S3.

1. Faça login AWS Management Console e abra o console do IAM em <https://console.aws.amazon.com/iam/>.
2. No painel de navegação, selecione Perfis e, em seguida, Criar perfil.
3. Em Selecionar entidade confiável, em Tipo de entidade confiável, escolha AWS service (Serviço da AWS).
4. Em Caso de uso, escolha Amazon EC2 e, em seguida, Próximo.
5. Em Adicionar permissões, marque a caixa de seleção do Acesso somente leitura do Amazon S3 na lista de políticas e, em seguida, selecione Próximo.
6. Insira um nome para o perfil e, em seguida, escolha Criar perfil. Lembre-se desse nome, pois você precisará dele quando executar sua instância do Amazon EC2.

## Executar uma instância do Amazon EC2 e especificar seu perfil do IAM

Você pode executar uma instância do Amazon EC2; com um perfil do IAM usando o console do Amazon EC2;

Siga as instruções para iniciar uma instância no Guia do usuário do [Amazon EC2](#) ou no [Guia do usuário](#) do [Amazon EC2](#).

Quando você chegar à página Review Instance Launch (Revisar ativação da instância), selecione Edit instance details (Editar detalhes da instância). Em Perfil do IAM, escolha o perfil do IAM criado por você anteriormente. Conclua o procedimento conforme indicado.

### Note

Você precisará criar ou usar um grupo de segurança e par de chaves para se conectar à instância.

Com esta configuração do IAM e do Amazon EC2, você pode implantar seu aplicativo na instância do Amazon EC2 e ter acesso de leitura ao serviço do Amazon S3.

## Conectar à instância do EC2

Conectar à instância do EC2 para que você possa transferir a amostra do aplicativo e, em seguida, executar o aplicativo. Você precisará do arquivo que contém a parte privada do par de chaves usado para executar a instância, ou seja, o arquivo PEM.

Você pode fazer isso seguindo o procedimento de conexão no Guia do usuário do [Amazon EC2](#) ou no [Guia do usuário](#) do [Amazon EC2](#). Ao conectar-se, faça isso de maneira que possa transferir arquivos da sua máquina de desenvolvimento para sua instância.

Se você estiver usando um AWS kit de ferramentas, geralmente também poderá se conectar à instância usando o kit de ferramentas. Para obter mais informações, consulte o Guia do usuário específico para o kit de ferramentas que você usa.

## Executar a amostra de aplicativo na instância do EC2

1. Copiar as aplicações do aplicativo da sua unidade local para sua instância.

Para obter informações sobre como transferir arquivos para sua instância, consulte o Guia do usuário do [Amazon EC2](#) ou o [Guia do usuário](#) do [Amazon EC2](#).

2. Inicie o aplicativo e verifique se ele é executado com os mesmos resultados da sua máquina de desenvolvimento.
3. (Opcional) Verifique se o aplicativo usa as credenciais fornecidas pelo perfil do IAM.
  - a. [Faça login no AWS Management Console e abra o console do Amazon EC2 em https://console.aws.amazon.com/ec2/](https://console.aws.amazon.com/ec2/).
  - b. Selecione a instância e desvincule o perfil do IAM pelo Ações, Configurações da instância, Anexar/Substituir perfil do IAM.
  - c. Execute o aplicativo novamente e confirme se ele retorna um erro de autorização.

## Referência de configurações

Os SDKs fornecem APIs específicas de linguagem para. Serviços da AWS Eles cuidam de parte do trabalho pesado necessário para fazer chamadas de API com sucesso, incluindo autenticação, comportamento de repetição e muito mais. Para fazer isso, os SDKs têm estratégias flexíveis para obter credenciais para usar em suas solicitações, manter as configurações a serem usadas com cada serviço e obter valores a serem usados nas configurações globais.

Você pode encontrar informações detalhadas sobre as definições de configuração nas seções a seguir:

- [AWS Provedores de credenciais padronizados de SDKs e ferramentas](#) – Provedores de credenciais comuns padronizados em vários SDKs.
- [AWS Recursos padronizados de SDKs e ferramentas](#) – Atributos comuns padronizados em vários SDKs.

## Criar clientes de serviço

Para acessar programaticamente Serviços da AWS, os SDKs usam uma classe/objeto cliente para cada um. AWS service (Serviço da AWS) Se seu aplicativo precisar acessar o Amazon EC2, por exemplo, seu aplicativo criará um objeto cliente do Amazon EC2 para interagir com esse serviço. Em seguida, você usa o cliente de serviço para fazer solicitações para esse AWS service (Serviço da AWS). Na maioria dos SDKs, um objeto de cliente de serviço é imutável, então você deve criar um novo cliente para cada serviço para o qual você faz solicitações e para fazer solicitações ao mesmo serviço usando uma configuração diferente.

## Precedência de configurações

As configurações globais definem atributos, provedores de credenciais e outras funcionalidades que são suportadas pela maioria dos SDKs e têm um amplo impacto nos Serviços da AWS. Todos os SDKs têm uma série de locais (ou fontes) que eles verificam para encontrar um valor para as configurações globais. A seguir está a configuração da precedência de pesquisa:

1. Qualquer configuração explícita definida no código ou no próprio cliente de serviço tem precedência sobre qualquer outra coisa.



- Algumas configurações podem ser definidas por operação e podem ser alteradas conforme necessário para cada operação que você invocar. Para o AWS CLI ou AWS Tools for PowerShell, eles assumem a forma de parâmetros por operação que você insere na linha de comando. Para um SDK, as atribuições explícitas podem assumir a forma de um parâmetro definido ao instanciar um AWS service (Serviço da AWS) cliente ou objeto de configuração ou, às vezes, ao chamar uma API individual.
2. Somente Java/Kotlin: a propriedade do sistema JVM para a configuração é verificada. Se estiver definido, esse valor é usado para configurar o cliente.
  3. A variável de ambiente está marcada. Se estiver definido, esse valor é usado para configurar o cliente.
  4. O SDK verifica a configuração no `credentials` arquivo compartilhado. Se estiver definido, o cliente o usará.
  5. O `config` arquivo compartilhado para a configuração. Se a configuração estiver presente, o SDK a usará.
    - A variável de `AWS_PROFILE` ambiente ou a propriedade do sistema `aws.profile` JVM pode ser usada para especificar qual perfil o SDK carrega.
  6. Qualquer valor padrão fornecido pelo próprio código-fonte do SDK é usado por último.

#### Note

Alguns SDKs e ferramentas podem ser verificados em uma ordem diferente. Além disso, alguns SDKs e ferramentas oferecem suporte a outros métodos de armazenamento e recuperação de parâmetros. Por exemplo, o AWS SDK for .NET suporta uma fonte adicional chamada [SDK Store](#). Para obter mais informações sobre provedores exclusivos de um SDK ou ferramenta, consulte o guia específico do SDK ou da ferramenta que você está usando.

A ordem determina quais métodos têm precedência e substituem outros. Por exemplo, se você configurar um perfil no arquivo `config` compartilhado, ele só será encontrado e usado depois que o SDK ou a ferramenta verificarem primeiro os outros lugares. Isso significa que, se você colocar uma configuração no arquivo `credentials`, ela será usada em vez de uma encontrada no arquivo `config`. Se você configurar uma variável de ambiente com uma configuração e um valor, ela substituirá essa configuração nos arquivos `credentials` e `config`. E, finalmente, uma configuração na operação individual (parâmetro da API ou parâmetro da linha de comando AWS CLI) ou no código substituiria todos os outros valores desse comando.

## Lista de configurações de arquivo **Config**

As configurações listadas na tabela a seguir podem ser atribuídas no AWS config arquivo compartilhado. Eles são globais e afetam a todos os Serviços da AWS. SDKs e ferramentas também podem oferecer suporte a configurações e variáveis de ambiente exclusivas. Para ver as configurações e as variáveis de ambiente suportadas somente por um SDK ou ferramenta individual, consulte esse SDK ou guia de ferramentas específico.

Nome da configuração	Detalhes
api_versions	<a href="#">Definições gerais de configuração</a>
aws_access_key_id	<a href="#">AWS teclas de acesso</a>
aws_secret_access_key	<a href="#">AWS teclas de acesso</a>
aws_session_token	<a href="#">AWS teclas de acesso</a>
ca_bundle	<a href="#">Definições gerais de configuração</a>
credential_process	<a href="#">Provedor de credenciais de processo</a>
credential_source	<a href="#">Assuma a função de provedor de credenciais</a>
defaults_mode	<a href="#">Padrões de configuração inteligente</a>
disable_request_compression	<a href="#">Compactação de solicitações</a>
duration_seconds	<a href="#">Assuma a função de provedor de credenciais</a>

Nome da configuração	Detalhes
ec2_metadata_service_endpoint	<a href="#">Provedor de credenciais IMDS</a>
ec2_metadata_service_endpoint_mode	<a href="#">Provedor de credenciais IMDS</a>
ec2_metadata_v1_disabled	<a href="#">Provedor de credenciais IMDS</a>
endpoint_discovery_enabled	<a href="#">Descoberta de endpoint</a>
endpoint_url	<a href="#">Endpoints específicos de serviço</a>
external_id	<a href="#">Assuma a função de provedor de credenciais</a>
ignore_configured_endpoint_urls	<a href="#">Endpoints específicos de serviço</a>
max_attempts	<a href="#">Comportamento de repetição</a>
metadata_service_num_attempts	<a href="#">Metadados da instância do Amazon EC2</a>
metadata_service_timeout	<a href="#">Metadados da instância do Amazon EC2</a>
mfa_serial	<a href="#">Assuma a função de provedor de credenciais</a>

Nome da configuração	Detalhes
output	<a href="#">Definições gerais de configuração</a>
parameter_validation	<a href="#">Definições gerais de configuração</a>
region	<a href="#">Região da AWS</a>
request_max_in_compression_size_bytes	<a href="#">Compactação de solicitações</a>
retry_mode	<a href="#">Comportamento de repetição</a>
role_arn	<a href="#">Assuma a função de provedor de credenciais</a>
role_session_name	<a href="#">Assuma a função de provedor de credenciais</a>
s3_disable_multiregion_access_points	<a href="#">Pontos de acesso de várias regiões do Amazon S3</a>
s3_use_arn_region	<a href="#">Pontos de acesso Amazon S3</a>
sdk_ua_app_id	<a href="#">ID do aplicativo</a>
source_profile	<a href="#">Assuma a função de provedor de credenciais</a>
sso_account_id	<a href="#">Provedor de credenciais do IAM Identity Center</a>
sso_region	<a href="#">Provedor de credenciais do IAM Identity Center</a>
sso_registration_scopes	<a href="#">Provedor de credenciais do IAM Identity Center</a>

Nome da configuração	Detalhes
sso_role_name	<a href="#">Provedor de credenciais do IAM Identity Center</a>
sso_start_url	<a href="#">Provedor de credenciais do IAM Identity Center</a>
sts_regional_endpoints	<a href="#">AWS STS Endpoints regionalizados</a>
use_dualstack_endpoint	<a href="#">Endpoints de pilha dupla e FIPS</a>
use_fips_endpoint	<a href="#">Endpoints de pilha dupla e FIPS</a>
web_identity_token_file	<a href="#">Assuma a função de provedor de credenciais</a>

## Lista de configurações de arquivo **Credentials**

As configurações listadas na tabela a seguir podem ser atribuídas no AWS `credentials` arquivo compartilhado. Eles são globais e afetam a todos os Serviços da AWS. SDKs e ferramentas também podem oferecer suporte a configurações e variáveis de ambiente exclusivas. Para ver as configurações e as variáveis de ambiente suportadas somente por um SDK ou ferramenta individual, consulte esse SDK ou guia de ferramentas específico.

Nome da configuração	Detalhes
aws_access_key_id	<a href="#">AWS teclas de acesso</a>
aws_secret_access_key	<a href="#">AWS teclas de acesso</a>
aws_session_token	<a href="#">AWS teclas de acesso</a>

## Lista de variáveis de ambiente

As variáveis de ambiente compatíveis com a maioria dos SDKs estão listadas na tabela a seguir. Eles são globais e afetam a todos Serviços da AWS. SDKs e ferramentas também podem oferecer suporte a configurações e variáveis de ambiente exclusivas. Para ver as configurações e as variáveis de ambiente suportadas somente por um SDK ou ferramenta individual, consulte esse SDK ou guia de ferramentas específico.

Nome da configuração	Detalhes
AWS_ACCESS_KEY_ID	<a href="#">AWS teclas de acesso</a>
AWS_CA_BUNDLE	<a href="#">Definições gerais de configuração</a>
AWS_CONFIG_FILE	<a href="#">Localização dos arquivos config e credentials compartilhados</a>
AWS_CONTAINER_AUTHORIZATION_TOKEN	<a href="#">Provedor de credenciais de contêiner</a>
AWS_CONTAINER_AUTHORIZATION_TOKEN_FILE	<a href="#">Provedor de credenciais de contêiner</a>
AWS_CONTAINER_CREDENTIALS_FULL_URI	<a href="#">Provedor de credenciais de contêiner</a>
AWS_CONTAINER_CREDENTIALS_RELATIVE_URI	<a href="#">Provedor de credenciais de contêiner</a>

Nome da configuração	Detalhes
AWS_DEFAULTS_MODE	<a href="#">Padrões de configuração inteligente</a>
AWS_DISABLE_REQUEST_COMPRESSION	<a href="#">Compactação de solicitações</a>
AWS_EC2_METADATA_DISABLED	<a href="#">Provedor de credenciais IMDS</a>
AWS_EC2_METADATA_SERVICE_ENDPOINT	<a href="#">Provedor de credenciais IMDS</a>
AWS_EC2_METADATA_SERVICE_ENDPOINT_MODE	<a href="#">Provedor de credenciais IMDS</a>
AWS_EC2_METADATA_V1_DISABLED	<a href="#">Provedor de credenciais IMDS</a>
AWS_ENABLE_ENDPOINT_DISCOVERY	<a href="#">Descoberta de endpoint</a>
AWS_ENDPOINT_URL	<a href="#">Endpoints específicos de serviço</a>
AWS_ENDPOINT_URL_<SERVICE>	<a href="#">Endpoints específicos de serviço</a>

Nome da configuração	Detalhes
AWS_IAM_ROLE_ARN	<a href="#">Assuma a função de provedor de credenciais</a>
AWS_IAM_ROLE_SESSION_NAME	<a href="#">Assuma a função de provedor de credenciais</a>
AWS_IGNORE_CONFIGURED_ENDPOINT_URLS	<a href="#">Endpoints específicos de serviço</a>
AWS_MAX_ATTEMPTS	<a href="#">Comportamento de repetição</a>
AWS_METADATA_SERVICE_NUM_ATTEMPTS	<a href="#">Metadados da instância do Amazon EC2</a>
AWS_METADATA_SERVICE_TIMEOUT	<a href="#">Metadados da instância do Amazon EC2</a>
AWS_PROFILE	<a href="#">Arquivos config e credentials compartilhados</a>
AWS_REGION	<a href="#">Região da AWS</a>
AWS_REQUEST_MIN_COMPRESSION_SIZE_BYTES	<a href="#">Compactação de solicitações</a>
AWS_RETRY_MODE	<a href="#">Comportamento de repetição</a>



Nome da configuração	Detalhes
AWS_S3_DISTRIBUTABLE_REGIONAL_ACCESS_POINTS	<a href="#">Pontos de acesso de várias regiões do Amazon S3</a>
AWS_S3_US_EARN_REGION	<a href="#">Pontos de acesso Amazon S3</a>
AWS_SDK_USER_APP_ID	<a href="#">ID do aplicativo</a>
AWS_SECRET_ACCESS_KEY	<a href="#">AWS teclas de acesso</a>
AWS_SESSION_TOKEN	<a href="#">AWS teclas de acesso</a>
AWS_SHARED_CREDENTIALS_FILE	<a href="#">Localização dos arquivos config e credentials compartilhados</a>
AWS_STS_REGIONAL_ENDPOINTS	<a href="#">AWS STS Endpoints regionalizados</a>
AWS_USE_DUALSTACK_ENDPOINT	<a href="#">Endpoints de pilha dupla e FIPS</a>
AWS_USE_FIPS_ENDPOINT	<a href="#">Endpoints de pilha dupla e FIPS</a>
AWS_WEB_IDENTITY_TOKEN_FILE	<a href="#">Assuma a função de provedor de credenciais</a>

## Lista de propriedades do sistema JVM

Você pode usar as seguintes propriedades do sistema JVM para o AWS SDK for Java e o AWS SDK para Kotlin (visando a JVM). Consulte [the section called “Como definir as propriedades do sistema JVM”](#) para obter instruções sobre como definir as propriedades do sistema JVM.

Nome da configuração	Detalhes
<code>aws.accessKeyId</code>	<a href="#">AWS teclas de acesso</a>
<code>aws.configFile</code>	<a href="#">Localização dos arquivos config e credentials compartilhados</a>
<code>aws.defaultsMode</code>	<a href="#">Padrões de configuração inteligente</a>
<code>aws.disableEc2MetadataV1</code>	<a href="#">Provedor de credenciais IMDS</a>
<code>aws.disableRequestCompression</code>	<a href="#">Compactação de solicitações</a>
<code>aws.ec2MetadataServiceEndpoint</code>	<a href="#">Provedor de credenciais IMDS</a>
<code>aws.ec2MetadataServiceEndpointMode</code>	<a href="#">Provedor de credenciais IMDS</a>
<code>aws.endpointDiscoveryEnabled</code>	<a href="#">Descoberta de endpoint</a>
<code>aws.endpointUrl</code>	<a href="#">Endpoints específicos de serviço</a>

Nome da configuração	Detalhes
<code>aws.endpointUrl&lt;ServiceName&gt;</code>	<a href="#">Endpoints específicos de serviço</a>
<code>aws.ignoreConfiguredEndpointUrls</code>	<a href="#">Endpoints específicos de serviço</a>
<code>aws.maxAttempts</code>	<a href="#">Comportamento de repetição</a>
<code>aws.profile</code>	<a href="#">Arquivos config e credentials compartilhados</a>
<code>aws.region</code>	<a href="#">Região da AWS</a>
<code>aws.requestMinCompressionSizeBytes</code>	<a href="#">Compactação de solicitações</a>
<code>aws.retryMode</code>	<a href="#">Comportamento de repetição</a>
<code>aws.roleArn</code>	<a href="#">Assuma a função de provedor de credenciais</a>
<code>aws.roleSessionName</code>	<a href="#">Assuma a função de provedor de credenciais</a>
<code>aws.s3DisableMultiRegionAccessPoints</code>	<a href="#">Pontos de acesso de várias regiões do Amazon S3</a>
<code>aws.s3UseArnRegion</code>	<a href="#">Pontos de acesso Amazon S3</a>

Nome da configuração	Detalhes
<code>aws.secretsAccessKey</code>	<a href="#">AWS teclas de acesso</a>
<code>aws.sessionToken</code>	<a href="#">AWS teclas de acesso</a>
<code>aws.sharedCredentialsFile</code>	<a href="#">Localização dos arquivos config e credentials compartilhados</a>
<code>aws.useDualstackEndpoint</code>	<a href="#">Endpoints de pilha dupla e FIPS</a>
<code>aws.useFipsEndpoint</code>	<a href="#">Endpoints de pilha dupla e FIPS</a>
<code>aws.userAgentAppId</code>	<a href="#">ID do aplicativo</a>
<code>aws.webIdentityTokenFile</code>	<a href="#">Assuma a função de provedor de credenciais</a>

## AWS Provedores de credenciais padronizados de SDKs e ferramentas

Muitos provedores de credenciais foram padronizados para padrões consistentes e para funcionar da mesma forma em muitos SDKs. Essa consistência aumenta a produtividade e a clareza ao codificar em vários SDKs. Todas as configurações podem ser substituídas no código. Para obter detalhes, consulte a API específica de seu SDK.

**⚠ Important**

Nem todos os SDKs oferecem suporte a todos os provedores, nem mesmo a todos os aspectos de um provedor.

## Tópicos

- [Cadeia de fornecedores de credenciais](#)
- [AWS chaves de acesso](#)
- [Assuma o perfil de provedor de credenciais](#)
- [Provedor de credenciais de contêiner](#)
- [Provedor de credencial do IAM Identity Center](#)
- [Provedor de credenciais IMDS](#)
- [Provedor de credenciais de processo](#)

## Cadeia de fornecedores de credenciais

Todos os SDKs têm uma série de locais (ou fontes) que eles verificam para encontrar credenciais válidas para usar para fazer uma solicitação a um AWS service (Serviço da AWS). Depois que as credenciais válidas são encontradas, a pesquisa é interrompida. Essa busca sistemática é chamada de cadeia de fornecedores de credencial padrão.

Embora a cadeia distinta usada por cada SDK varie, elas geralmente incluem fontes como as seguintes:

Provedor de credencial	Descrição
<a href="#">AWS chaves de acesso</a>	AWS chaves de acesso para um usuário do IAM (como <code>AWS_ACCESS_KEY_ID</code> e <code>AWS_SECRET_ACCESS_KEY</code> ).
<a href="#">Federar com identidade web ou OpenID Connect</a> - Assumir a função de provedor de credenciais	Fazer login usando um provedor de identidades (IdP) externo conhecido, como Login with Amazon, Facebook, Google ou qualquer outro IdP compatível com OpenID Connect (OIDC). Assuma as permissões de uma função

Provedor de credencial	Descrição
	do IAM usando um token de identidade da web de AWS Security Token Service (AWS STS).
<a href="#">Provedor de credencial do IAM Identity Center</a>	Obtenha credenciais de AWS IAM Identity Center.
<a href="#">Assuma o perfil de provedor de credenciais</a>	Tenha acesso a outros recursos assumindo as permissões de um perfil do IAM. (Recupere e use credenciais temporárias para uma função).
<a href="#">Provedor de credenciais de contêiner</a>	Credenciais do Amazon Elastic Container Service (Amazon ECS) e do Amazon Elastic Kubernetes Service (Amazon EKS). O provedor de credenciais de contêiner busca credenciais para o aplicativo em contêiner do cliente.
<a href="#">Provedor de credenciais de processo</a>	Provedores de credenciais personalizados. Obtenha suas credenciais de uma fonte ou processo externo, incluindo o IAM Roles Anywhere.
<a href="#">Provedor de credenciais IMDS</a>	Credenciais do perfil de instância do Amazon Elastic Compute Cloud (Amazon EC2). Associe um perfil do IAM a cada uma das suas instâncias do EC2. As credenciais temporárias para essa função estão disponíveis para o código em execução na instância. As credenciais são fornecidas por meio do serviço de metadados do Amazon EC2.

Para cada etapa da cadeia, há várias maneiras de atribuir valores de configuração. Os valores de configuração especificados no código sempre têm precedência. No entanto, também existem [Variáveis de ambiente](#) e [Arquivos config e credentials compartilhados](#). Para ter mais informações, consulte [Precedência de configurações](#).

## AWS chaves de acesso

### Warning

Para evitar riscos de segurança, não use usuários do IAM para autenticação ao desenvolver software com propósito específico ou trabalhar com dados reais. Em vez disso, use federação com um provedor de identidade, como [AWS IAM Identity Center](#).

AWS as chaves de acesso de um usuário do IAM podem ser usadas como suas AWS credenciais. O AWS SDK usa automaticamente essas AWS credenciais para assinar solicitações de API AWS, para que suas cargas de trabalho possam acessar seus AWS recursos e dados de forma segura e conveniente. É recomendável sempre usar o `aws_session_token` para que as credenciais sejam temporárias e não sejam mais válidas após expirarem. Não é recomendável usar credenciais de longo prazo.

### Note

Se AWS não conseguir atualizar essas credenciais temporárias, AWS poderá estender a validade das credenciais para que suas cargas de trabalho não sejam afetadas.

O `AWS credentials` arquivo compartilhado é o local recomendado para armazenar informações de credenciais porque está fora dos diretórios de origem do aplicativo e separado das configurações específicas do SDK do arquivo compartilhado. `config`

Para saber mais sobre AWS credenciais e o uso de chaves de acesso, consulte [Credenciais AWS de segurança](#) e [Gerenciamento de chaves de acesso para usuários do IAM no Guia](#) do usuário do IAM.

Configure esta funcionalidade usando o seguinte:

**aws\_access\_key\_id**- configuração de AWS **config** arquivo compartilhado,  
**aws\_access\_key\_id**- configuração de AWS **credentials** arquivo compartilhado (método recomendado), **AWS\_ACCESS\_KEY\_ID** - variável de ambiente, **aws.accessKeyId**- Propriedade do sistema JVM: somente Java/Kotlin

Especifica a chave de AWS acesso usada como parte das credenciais para autenticar o usuário.

**aws\_secret\_access\_key**- configuração de AWS **config** arquivo compartilhado,  
**aws\_secret\_access\_key**- configuração de AWS **credentials** arquivo compartilhado (método recomendado), **AWS\_SECRET\_ACCESS\_KEY** - variável de ambiente, **aws.secretAccessKey**- Propriedade do sistema JVM: somente Java/Kotlin

Especifica a chave AWS secreta usada como parte das credenciais para autenticar o usuário.

**aws\_session\_token**- configuração de AWS **config** arquivo compartilhado,  
**aws\_session\_token**- configuração de AWS **credentials** arquivo compartilhado (método recomendado), **AWS\_SESSION\_TOKEN** - variável de ambiente, **aws.sessionToken**- Propriedade do sistema JVM: somente Java/Kotlin

Especifica um token de AWS sessão usado como parte das credenciais para autenticar o usuário. Você recebe esse valor como parte das credenciais temporárias retornadas por solicitações bem-sucedidas para assumir uma função. Um token de sessão só será necessário se você especificar manualmente credenciais de segurança temporárias. No entanto, recomendamos que você use sempre credenciais de segurança temporárias em vez de credenciais de longo prazo. Para obter recomendações de segurança, consulte [Melhores práticas de segurança no IAM](#).

Para obter instruções sobre como obter esses valores, consulte [Autenticar usando credenciais de curto prazo](#).

Exemplo de configuração desses valores necessários no config ou credentials arquivo:

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

Exemplo de configuração de variáveis de ambiente para Linux/macOS por meio da linha de comando:

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

Exemplo do Windows de configuração de variáveis de ambiente por meio da linha de comando:

```
setx AWS_ACCESS_KEY_ID AKIAIOSFODNN7EXAMPLE
```



```
setx AWS_SECRET_ACCESS_KEY wJa1rXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
setx
AWS_SESSION_TOKEN AqoEXAMPLEH4aoAH0gNCAPy...truncated...zrkuWJ0gQs8IZZaIv2BXIa2R40Lgk
```

## Compatibilidade com AWS SDKs

Os SDKs a seguir são compatíveis com os recursos e configurações descritos neste tópico. Quaisquer exceções parciais estão anotadas. Todas as configurações de propriedade do sistema JVM são suportadas pelo AWS SDK for Java e pelo AWS SDK para Kotlin único.

SDK	Compatível	Notas ou mais informações
<a href="#">AWS CLI v2</a>	Sim	
<a href="#">SDK para C++</a>	Sim	arquivo compartilhado config não suportado.
<a href="#">SDK para Go V2 (1.x)</a>	Sim	
<a href="#">SDK for Go 1.x (V1)</a>	Sim	Para usar as configurações do arquivo config compartilhado, você deve ativar o carregamento do arquivo de configuração; consulte <a href="#">Sessões</a> .
<a href="#">SDK para Java 2.x</a>	Sim	
<a href="#">SDK para Java 1.x</a>	Sim	
<a href="#">SDK para 3.x JavaScript</a>	Sim	
<a href="#">SDK para 2.x JavaScript</a>	Sim	
<a href="#">SDK for Kotlin</a>	Sim	
<a href="#">SDK for .NET 3.x</a>	Sim	As variáveis de ambiente não são compatíveis.
<a href="#">SDK para PHP 3.x</a>	Sim	
<a href="#">SDK for Python (Boto3)</a>	Sim	
<a href="#">SDK for Ruby 3.x</a>	Sim	

SDK	C	Notas ou mais informações
<a href="#">SDK for Rust</a>	Sim	
<a href="#">Ferramentas para PowerShell</a>	Sim	As variáveis de ambiente não são compatíveis.

## Assuma o perfil de provedor de credenciais

Assumir um perfil envolve o uso de um conjunto de credenciais temporárias de segurança para acessar AWS recursos aos quais você talvez não tenha acesso de outra forma. Essas credenciais de segurança temporárias consistem em um ID de chave de acesso, uma chave de acesso secreta e um token de segurança.

Para configurar seu SDK ou ferramenta para assumir um perfil, você deve primeiro criar ou identificar um perfil específico a ser assumido. [Os perfis do IAM são identificados exclusivamente por um perfil do nome do recurso da Amazon \(ARN\)](#). Os perfis estabelecem as relações de confiança com uma outra entidade. A entidade confiável que usa a função pode ser uma AWS service (Serviço da AWS), outra Conta da AWS, um provedor de identidade da web ou uma federação OIDC ou SAML.

Depois que perfil do IAM for identificado, se você tiver a confiança desse perfil, poderá configurar seu SDK ou ferramenta para usar as permissões concedidas pelo perfil. Para fazer isso, execute as configurações a seguir.

Para obter orientação sobre como começar a usar essas configurações, consulte este guia [Assumir uma função](#).

### Assuma as configurações do provedor de credenciais do perfil

Configure esta funcionalidade usando o seguinte:

**credential\_source**- configuração de AWS **config** arquivo compartilhado

Usado em instâncias ou contêineres do Amazon EC2 ou Amazon Elastic Container Service para especificar onde o SDK ou ferramenta pode encontrar credenciais que tem permissão para assumir o perfil que você especificar com o parâmetro `role_arn`.

Valor padrão: nenhum

Valores válidos:

- **Ambiente:** especifica que o SDK ou a ferramenta deve recuperar credenciais de origem de variáveis de ambiente [AWS\\_ACCESS\\_KEY\\_ID](#) e [AWS\\_SECRET\\_ACCESS\\_KEY](#).
- **Ec2 InstanceMetadata** — Especifica que o SDK ou a ferramenta deve usar a [função do IAM anexada ao perfil da instância do EC2 para obter as](#) credenciais de origem.
- **EcsContainer**— Especifica que o SDK ou a ferramenta deve usar a [função do IAM anexada ao contêiner do ECS para obter as](#) credenciais de origem.

Não é possível especificar `credential_source` e `source_profile` no mesmo perfil.

Exemplo de configuração em um `config` arquivo para indicar que as credenciais devem ser provenientes do Amazon EC2:

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

**duration\_seconds**- configuração de AWS **config** arquivo compartilhado

Especifica a duração máxima da sessão da função, em segundos.

Esta configuração se aplica somente quando o perfil especifica assumir uma função.

Valor padrão: 3.600 segundos (uma hora)

Valores válidos: O valor pode variar de 900 segundos (15 minutos) até o valor configurado de duração máxima da sessão para o perfil (que pode ser até 43200, ou 12 horas). Para obter mais informações, consulte [Exibir a configuração de duração máxima da sessão para um perfil](#) no Guia do usuário do IAM.

Exemplo de configuração em um arquivo `config`:

```
duration_seconds = 43200
```

**external\_id**- configuração de AWS **config** arquivo compartilhado

Especifica um identificador exclusivo que é usado por terceiros para assumir uma função em suas contas de clientes.

Esta configuração se aplica somente quando o perfil especifica assumir uma função e a política de confiança do perfil exige um valor para `ExternalId`. O valor é mapeado para o parâmetro

ExternalId que é passado para a operação AssumeRole quando o perfil especifica uma função.

Valor padrão: nenhum.

Valores válidos: consulte [Como usar uma ID externa ao conceder acesso aos seus AWS recursos a terceiros](#) no Guia do usuário do IAM.

Exemplo de configuração em um arquivo config:

```
external_id = unique_value_assigned_by_3rd_party
```

**mfa\_serial**- configuração de AWS **config** arquivo compartilhado

Especifica a identificação ou o número de série de um dispositivo de autenticação multifator (MFA) que o usuário deve usar ao assumir um perfil.

Obrigatório ao assumir um perfil em que a política de confiança para o perfil inclui uma condição que exige autenticação de MFA.

Valor padrão: nenhum.

Valores válidos: O valor pode ser um número de série de um dispositivo de hardware (como GAHT12345678) ou um nome do recurso da Amazon (ARN) de um dispositivo MFA virtual. Para obter mais informações sobre MFA, consulte [Configuração de acesso à API protegido por MFA](#) no Guia do usuário do IAM.

Exemplo de configuração em um arquivo config:

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

**role\_arn**- configuração de AWS **config** arquivo compartilhado, **AWS\_IAM\_ROLE\_ARN** - variável de ambiente, **aws.roleArn**- Propriedade do sistema JVM: somente Java/Kotlin

Especifica o nome do recurso da Amazon (ARN) de um perfil do IAM que você deseja usar para realizar operações solicitadas usando esse perfil.

Valor padrão: nenhum.

Valores válidos: O valor deve ser o ARN de um perfil do IAM, formatado da seguinte forma:  
`arn:aws:iam::account-id:role/role-name`

Além disso, você também deve especificar uma das seguintes configurações:

- `source_profile`— Identificar outro perfil a ser usado para encontrar credenciais que tenham permissão para assumir a função nesse perfil.
- `credential_source`— Usar credenciais identificadas pelas variáveis de ambiente atuais ou credenciais anexadas a um perfil de instância do Amazon EC2 ou a uma instância de contêiner do Amazon ECS.
- `web_identity_token_file`— Usar provedores de identidades públicas ou qualquer provedor de identidades compatível com OpenID Connect (OIDC) para usuários que foram autenticados em uma aplicação móvel ou uma aplicação web.

**role\_session\_name**- configuração de AWS **config** arquivo compartilhado,

**AWS\_IAM\_ROLE\_SESSION\_NAME** - variável de ambiente, **aws.roleSessionName**- Propriedade do sistema JVM: somente Java/Kotlin

Especifica o nome a ser associado à sessão da função. Este nome aparece nos logs do AWS CloudTrail para entradas associadas a esta sessão, que pode ser útil em uma auditoria.

Valor padrão: um parâmetro opcional. Se você não fornecer este valor, um nome de sessão será gerado automaticamente se o perfil assumir uma função.

Valores válidos: fornecidos ao `RoleSessionName` parâmetro quando a AWS API AWS CLI ou chama a `AssumeRole` operação (ou operações como a `AssumeRoleWithWebIdentity` operação) em seu nome. O valor se torna parte da função assumida do usuário Amazon Resource Name (ARN) que você pode consultar e aparece como parte das entradas de CloudTrail registro das operações invocadas por esse perfil.

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

Exemplo de configuração em um arquivo `config`:

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

**source\_profile**- configuração de AWS **config** arquivo compartilhado

Especifica outro perfil cujas credenciais são usadas para assumir o perfil especificado pela configuração `role_arn` no perfil original. Para entender como os perfis são usados no compartilhamento AWS `config` e nos `credentials` arquivos, consulte [Arquivos `config` e `credentials` compartilhados](#).



Esta configuração permite a autenticação usando provedores de federação de identidade da web, como [Google](#), [Facebook](#) e [Amazon](#), entre muitos outros. O SDK ou a ferramenta do desenvolvedor carrega o conteúdo deste arquivo e o transmite como argumento `WebIdentityToken` quando chama a operação `AssumeRoleWithWebIdentity` em seu nome.

Valor padrão: nenhum.

Valores válidos: este valor deve ser um nome de caminho e de arquivo. O arquivo deve conter um token de acesso OAuth 2.0 ou token de OpenID Connect fornecido por um provedor de identidade a você. Os caminhos relativos são tratados como relativos ao diretório de trabalho do processo.

## Compatibilidade com AWS SDKs

Os SDKs a seguir são compatíveis com os recursos e configurações descritos neste tópico. Quaisquer exceções parciais estão anotadas. Todas as configurações de propriedade do sistema JVM são suportadas pelo AWS SDK for Java e pelo AWS SDK para Kotlin único.

SDK	C	Notas ou mais informações
<a href="#">AWS CLI v2</a>	Sim	
<a href="#">SDK for C++</a>	Parci	<code>credential_source</code> não suportado. <code>duration_seconds</code> não suportado. <code>mfa_serial</code> não suportado.
<a href="#">SDK for Go V2 (1.x)</a>	Sim	
<a href="#">SDK for Go 1.x (V1)</a>	Sim	Para usar as configurações do arquivo <code>config</code> compartilhado, você deve ativar o carregamento do arquivo de configuração; consulte <a href="#">Sessões</a> .
<a href="#">SDK para Java 2.x</a>	Parci	<code>mfa_serial</code> incompatível. Use <code>AWS_ROLE_ARN</code> em vez de <code>AWS_IAM_ROLE_ARN</code> . Use <code>AWS_ROLE_SESSION_NAME</code> em vez de <code>AWS_IAM_ROLE_SESSION_NAME</code> .
<a href="#">SDK para Java 1.x</a>	Parci	<code>mfa_serial</code> incompatível. As propriedades do sistema JVM não são suportadas.

SDK	Compatível	Notas ou mais informações
<a href="#">SDK para 3.x JavaScript</a>	Sim	
<a href="#">SDK para 2.x JavaScript</a>	Parcial	<code>credential_source</code> incompatível.
<a href="#">SDK for Kotlin</a>	Sim	Use <code>AWS_ROLE_ARN</code> em vez de <code>AWS_IAM_ROLE_ARN</code> . Use <code>AWS_ROLE_SESSION_NAME</code> em vez de <code>AWS_IAM_ROLE_SESSION_NAME</code> .
<a href="#">SDK for .NET 3.x</a>	Sim	
<a href="#">SDK for PHP 3.x</a>	Sim	
<a href="#">SDK for Python (Boto3)</a>	Sim	
<a href="#">SDK for Ruby 3.x</a>	Sim	
<a href="#">SDK for Rust</a>	Sim	
<a href="#">Ferramentas para PowerShell</a>	Sim	

## Provedor de credenciais de contêiner

O provedor de credenciais de contêiner busca credenciais para o aplicativo em contêiner do cliente. Este provedor de credenciais é útil para clientes do Amazon Elastic Container Service (Amazon ECS) e do Amazon Elastic Kubernetes Service (Amazon EKS). Os SDKs tentam carregar as credenciais do endpoint HTTP especificado por meio de uma solicitação GET.

Se você usa o Amazon ECS, recomendamos que você use um perfil do IAM de tarefa para melhorar o isolamento, a autorização e a auditabilidade das credenciais. Quando configurado, o Amazon ECS define a variável de ambiente `AWS_CONTAINER_CREDENTIALS_RELATIVE_URI` que os SDKs e as ferramentas usam para obter credenciais. Para configurar o Amazon ECS para essa funcionalidade, consulte a [Função do IAM de tarefa](#) no Amazon Elastic Container Service Developer Guide.

Se você usa o Amazon EKS, recomendamos usar o Amazon EKS Pod Identity para melhorar o isolamento de credenciais, privilégios mínimos, auditabilidade, operação independente,



reutilização e escalabilidade. Tanto seu Pod quanto um perfil do IAM estão associados a uma conta de serviço do Kubernetes para gerenciar as credenciais dos seus aplicativos. Para saber mais sobre o Amazon EKS Pod Identity, consulte [Amazon EKS Pod Identities](#) no Guia do usuário do Amazon EKS. Quando configurado, o Amazon EKS define as variáveis de ambiente `AWS_CONTAINER_CREDENTIALS_FULL_URI` e `AWS_CONTAINER_AUTHORIZATION_TOKEN_FILE` que os SDKs e as ferramentas usam para obter credenciais. Para obter informações de configuração, consulte [Configurar o Amazon EKS Pod Identity Agent](#) no Guia do usuário do Amazon EKS ou o [Amazon EKS Pod Identity simplifica as permissões do IAM para aplicativos em clusters do Amazon EKS](#) no site do AWS blog.

Configure esta funcionalidade usando o seguinte:

### **AWS\_CONTAINER\_CREDENTIALS\_FULL\_URI** - variável de ambiente

Contém o endpoint de URL HTTP relativo para o SDK usar ao fazer uma solicitação de credenciais. Isso inclui o esquema e o host.

Valor padrão: Nenhum.

Valores válidos: URI válido.

Nota: Essa configuração é uma alternativa para `AWS_CONTAINER_CREDENTIALS_RELATIVE_URI` e só será usada se `AWS_CONTAINER_CREDENTIALS_RELATIVE_URI` não estiver definida.

Exemplo de configuração de variáveis de ambiente no Linux/macOS por meio da linha de comando:

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

ou

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

### **AWS\_CONTAINER\_CREDENTIALS\_RELATIVE\_URI** - variável de ambiente

Contém o endpoint de URL HTTP completo para o SDK usar ao fazer uma solicitação de credenciais. O valor é anexado ao nome de host padrão do Amazon ECS de `169.254.170.2`.

Valor padrão: Nenhum.

Valores válidos: URI relativo válido.

Exemplo de configuração de variáveis de ambiente no Linux/macOS por meio da linha de comando:

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

### **AWS\_CONTAINER\_AUTHORIZATION\_TOKEN** - variável de ambiente

Especifica o token de autorização em texto sem formatação. Se essa variável for definida, o SDK definirá o cabeçalho de autorização na solicitação HTTP com o valor da variável de ambiente.

Valor padrão: Nenhum.

Valores válidos: String.

Nota: Essa configuração é uma alternativa para `AWS_CONTAINER_AUTHORIZATION_TOKEN_FILE` e só será usada se `AWS_CONTAINER_AUTHORIZATION_TOKEN_FILE` não estiver definida.

Exemplo de configuração de variáveis de ambiente para Linux/macOS por meio da linha de comando:

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

### **AWS\_CONTAINER\_AUTHORIZATION\_TOKEN\_FILE** - variável de ambiente

Especifica um caminho de arquivo absoluto para um arquivo que contém o token de autorização em texto simples.

Valor padrão: Nenhum.

Valores válidos: String.

Exemplo de configuração de variáveis de ambiente para Linux/macOS por meio da linha de comando:

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

## Compatibilidade com AWS SDKs

Os SDKs a seguir são compatíveis com os recursos e configurações descritos neste tópico. Quaisquer exceções parciais estão anotadas. Todas as configurações de propriedade do sistema JVM são suportadas pelo AWS SDK for Java e pelo AWS SDK para Kotlin único.

SDK	Compatível	Notas ou mais informações
<a href="#">AWS CLI v2</a>	Sim	
<a href="#">SDK para C++</a>	Sim	
<a href="#">SDK for Go V2 (1.x)</a>	Sim	
<a href="#">SDK for Go 1.x (V1)</a>	Sim	
<a href="#">SDK para Java 2.x</a>	Sim	
<a href="#">SDK para Java 1.x</a>	Parcial	Amazon EKS Pod Identity e <code>AWS_CONTAINER_AUTHORIZATION_TOKEN_FILE</code> não suportados.
<a href="#">SDK para 3.x JavaScript</a>	Sim	
<a href="#">SDK para 2.x JavaScript</a>	Sim	
<a href="#">SDK for Kotlin</a>	Sim	
<a href="#">SDK for .NET 3.x</a>	Sim	
<a href="#">SDK for PHP 3.x</a>	Sim	
<a href="#">SDK for Python (Boto3)</a>	Sim	
<a href="#">SDK for Ruby 3.x</a>	Sim	
<a href="#">SDK for Rust</a>	Sim	
<a href="#">Ferramentas para PowerShell</a>	Sim	

## Provedor de credencial do IAM Identity Center

Esse mecanismo de autenticação é usado AWS IAM Identity Center para obter acesso de login único (SSO) ao seu código Serviços da AWS .

### Note

Na documentação da API do AWS SDK, o provedor de credenciais do IAM Identity Center é chamado de provedor de credenciais SSO.

Depois de habilitar o IAM Identity Center, você define um perfil para suas configurações no seu AWS `config` arquivo compartilhado. Este perfil é usado para se conectar ao portal de acesso do IAM Identity Center. Quando um usuário se autentica com sucesso no IAM Identity Center, o portal retorna credenciais de curto prazo para o perfil do IAM associado a esse usuário. Para saber como o SDK obtém credenciais temporárias da configuração e as usa para AWS service (Serviço da AWS) solicitações, consulte [Entenda a autenticação do IAM Identity Center](#)

Há duas maneiras de configurar o IAM Identity Center por meio do arquivo `config`:

- Configuração do provedor de token SSO (recomendada) — Durações de sessão estendidas.
- Configuração legada não atualizável — usa uma sessão fixa de oito horas.

Em ambas as configurações, você precisa entrar novamente quando sua sessão expirar.

Para definir durações de sessão personalizadas, você deve usar a configuração do provedor de token SSO.

Os dois guias a seguir contêm informações adicionais sobre o IAM Identity Center:

- [AWS IAM Identity Center Guia do usuário](#)
- [AWS IAM Identity Center Referência da API do portal](#)

## Pré-requisitos

É necessário primeiro habilitar o IAM Identity Center. Para obter detalhes sobre como habilitar a autenticação do IAM Identity Center, consulte [Conceitos básicos](#) no Guia do usuário do AWS IAM Identity Center .

Como alternativa, siga as [Autenticação do IAM Identity Center](#) instruções deste guia. Estas instruções fornecem orientação completa, desde a habilitação do IAM Identity Center até a conclusão da configuração necessária de `config` arquivos compartilhados que segue aqui.

## Configuração do provedor de token do SSO

### Note

Para usar o AWS CLI para criar essa configuração para você, consulte [Configurar seu perfil com o `aws configure sso` assistente](#) no AWS CLI.

Quando você usa a configuração do provedor de token SSO, seu AWS SDK ou ferramenta atualiza automaticamente sua sessão até o período estendido da sessão. Para obter mais informações sobre a duração e a duração máxima da sessão, consulte [Configurar a duração da sessão do portal de AWS acesso e dos aplicativos integrados do IAM Identity Center](#) no Guia AWS IAM Identity Center do usuário.

A `sso-session` seção do `config` arquivo é usada para agrupar variáveis de configuração para adquirir tokens de acesso SSO, que podem então ser usados para adquirir AWS credenciais. Para obter mais detalhes sobre a formatação de seções em um arquivo `config`, consulte [Formato do arquivo de configuração](#).

Defina uma seção de `sso-session` e associe-a a um perfil. `sso_region` e `sso_start_url` devem ser definidos na seção `sso-session`. Normalmente, `sso_account_id` e `sso_role_name` deve ser definido na `profile` seção para que o SDK possa solicitar AWS credenciais.

### Note

Para saber mais detalhes sobre como os SDKs e as ferramentas usam e atualizam as credenciais usando esta configuração, consulte [Entenda a autenticação do IAM Identity Center](#).

O exemplo a seguir configura o SDK para solicitar credenciais do IAM Identity Center. Ele também oferece suporte à atualização automática de tokens.

```
[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
sso_registration_scopes = sso:account:access
```

Você pode reutilizar `sso-session` configurações em vários perfis.

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

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

[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
```

No entanto, `sso_account_id` e `sso_role_name` não são necessários para todos os cenários de configuração do token do SSO. Se seu aplicativo usa apenas Serviços da AWS essa autenticação de portador de suporte, AWS as credenciais tradicionais não são necessárias. A autenticação do portador é um esquema de autenticação HTTP que usa tokens de segurança chamados tokens de portador. Nesse cenário, `sso_account_id` e `sso_role_name` não são obrigatórios. Consulte o guia individual AWS service (Serviço da AWS) para determinar se ele suporta a autorização do token do portador.

Os escopos de registro são configurados como parte de `umsso-session`. O escopo é um mecanismo no OAuth 2.0 para limitar o acesso de um aplicativo à conta de um usuário. Um aplicativo pode solicitar um ou mais escopos, e o token de acesso emitido para o aplicativo está limitado aos escopos concedidos. Esses escopos definem as permissões solicitadas para serem autorizadas para o cliente OIDC registrado e os tokens de acesso recuperados pelo cliente. Para obter as opções de escopo de acesso compatíveis, consulte [Escopos de acesso](#) no Guia do usuário do AWS IAM Identity Center . O exemplo a seguir define `sso_registration_scopes` para fornecer acesso para listas de contas/perfis.

```
[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
```

O token de autenticação é armazenado em cache no disco sob o diretório `~/.aws/sso/cache` com um nome de arquivo baseado no nome da sessão.

## Configuração herdada não atualizável

A atualização automática de tokens não é compatível usando a configuração herdada não atualizável. Em vez disso, recomendamos usar [Configuração do provedor de token do SSO](#).

Para usar a configuração legada não atualizável, você deve especificar as seguintes configurações no seu perfil:

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

Você especifica o portal do usuário para um perfil com as configurações `sso_start_url` e `sso_region`. Você especifica as permissões com as configurações `sso_account_id` e `sso_role_name`.

O exemplo a seguir define os quatro valores necessários no arquivo `config`.

```
[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
```

O token de autenticação é armazenado em cache no disco sob o diretório `~/.aws/sso/cache` com um nome de arquivo baseado no `sso_start_url`.

## Configurações do provedor de credenciais do IAM Identity Center

Configure esta funcionalidade usando o seguinte:

**sso\_start\_url**- configuração de AWS **config** arquivo compartilhado

O URL que aponta para o portal de acesso ao IAM Identity Center da sua organização. Para obter mais informações sobre o portal de acesso do IAM Identity Center, consulte [Usando o portal de AWS acesso](#) no Guia AWS IAM Identity Center do usuário.

Para encontrar esse valor, abra o [console do IAM Identity Center](#), visualize o painel e encontre a URL do portal de AWS acesso.

**sso\_region**- configuração de AWS **config** arquivo compartilhado

O Região da AWS que contém o host do portal do IAM Identity Center; ou seja, a região que você selecionou antes de ativar o IAM Identity Center. Isso é independente da sua AWS região padrão e pode ser diferente.

Para obter uma lista completa dos Regiões da AWS e de seus códigos, consulte [Endpoints regionais](#) no Referência geral da Amazon Web Services. Para encontrar esse valor, abra o [console do IAM Identity Center](#), visualize o painel e encontre a região.

**sso\_account\_id**- configuração de AWS **config** arquivo compartilhado

O ID numérico do Conta da AWS que foi adicionado por meio do AWS Organizations serviço para uso na autenticação.

Para ver a lista de contas disponíveis, acesse o [console do IAM Identity Center](#) e abra a página de Contas da AWS. Você também pode ver a lista de contas disponíveis usando o método [ListAccounts](#)API na Referência da API do AWS IAM Identity Center Portal. Por exemplo, você pode chamar o AWS CLI método [list-accounts](#).

**sso\_role\_name**- configuração de AWS **config** arquivo compartilhado

O nome de um conjunto de permissões provisionado como um perfil do IAM que define as permissões resultantes do usuário. A função deve existir no Conta da AWS especificado por `sso_account_id`. Use o nome do perfil, não o nome do recurso da Amazon (ARN) do perfil.

Os conjuntos de permissões têm políticas do IAM e políticas de permissões personalizadas anexadas a eles e definem o nível de acesso que os usuários têm às suas Contas da AWS atribuídas.

Para ver a lista de conjuntos de permissões disponíveis por Conta da AWS, acesse o [console do IAM Identity Center](#) e abra a Contas da AWS página. Escolha o nome correto do conjunto de permissões listado na Contas da AWS tabela. Você também pode ver a lista de conjuntos de



permissões disponíveis usando o método [ListAccountRoles](#) API na Referência da API do AWS IAM Identity Center Portal. Por exemplo, você pode chamar o AWS CLI método [list-account-roles](#).

### **sso\_registration\_scopes**- configuração de AWS **config** arquivo compartilhado

Uma lista delimitada por vírgulas de escopos a serem autorizados para `sso-session`. Os escopos autorizam o acesso aos endpoints autorizados portadores do token do IAM Identity Center. Um escopo mínimo de `sso:account:access` deve ser concedido para recuperar um token de atualização do serviço IAM Identity Center. Para ver os strings de caracteres do escopo de acesso compatíveis, consulte [Escopos de acesso](#) no Guia do usuário AWS IAM Identity Center. Esta configuração não é aplicável à configuração legada não atualizável. Os tokens emitidos usando a configuração legada estão limitados ao escopo `sso:account:access` implícito.

## Compatibilidade com AWS SDKs

Os SDKs a seguir são compatíveis com os recursos e configurações descritos neste tópico. Quaisquer exceções parciais estão anotadas. Todas as configurações de propriedade do sistema JVM são suportadas pelo AWS SDK for Java e pelo AWS SDK para Kotlin único.

SDK	C	Notas ou mais informações
<a href="#">AWS CLI v2</a>	Sim	
<a href="#">SDK para C++</a>	Sim	
<a href="#">SDK for Go V2 (1.x)</a>	Sim	
<a href="#">SDK for Go 1.x (V1)</a>	Sim	Para usar as configurações do arquivo <code>config</code> compartilhado, você deve ativar o carregamento do arquivo de configuração; consulte <a href="#">Sessões</a> .
<a href="#">SDK para Java 2.x</a>	Sim	Valores de configuração também compatíveis no arquivo <code>credentials</code> .
<a href="#">SDK para Java 1.x</a>	Não	
<a href="#">SDK para 3.x JavaScript</a>	Sim	
<a href="#">SDK para 2.x JavaScript</a>	Sim	

SDK	Compatibilidade	Notas ou mais informações
<a href="#">SDK for Kotlin</a>	Sim	
<a href="#">SDK for .NET 3.x</a>	Sim	
<a href="#">SDK for PHP 3.x</a>	Sim	
<a href="#">SDK for Python (Boto3)</a>	Sim	
<a href="#">SDK for Ruby 3.x</a>	Sim	
<a href="#">SDK for Rust</a>	Parcial	Somente configuração herdada não atualizável.
<a href="#">Ferramentas para PowerShell</a>	Sim	

## Provedor de credenciais IMDS

O serviço de metadados de instância (IMDS) fornece dados sobre sua instância que é possível usar para configurar ou gerenciar a instância em execução. Para obter mais informações sobre os dados disponíveis, consulte [Metadados da instância e dados do usuário](#) no Guia do usuário do Amazon EC2 ou [Metadados da instância e dados do usuário no Guia do usuário do Amazon EC2](#). O Amazon EC2 fornece um endpoint local disponível para instâncias que podem fornecer várias informações para a instância. Se a instância tiver uma função anexada, ela poderá fornecer um conjunto de credenciais válidas para essa função. Os SDKs podem usar esse endpoint para resolver credenciais como parte de sua [cadeia de fornecedores de credenciais padrão](#). O serviço de metadados de instância versão 2 (IMDSv2), uma versão mais segura do IMDS que usa um token de sessão, é usado por padrão. Se ele falhar devido a uma condição que não pode ser repetida (códigos de erro HTTP 403, 404, 405), o IMDSv1 será usado como fallback.

Configure essa funcionalidade usando o seguinte:

**AWS\_EC2\_METADATA\_DISABLED**: variável de ambiente

Se o serviço de metadados de instância (IMDS) do Amazon EC2 (IMDS) deve ou não ser usado para obter credenciais.


Valor padrão: `false`.

Valores válidos:

- **true** – Não use o IMDS para obter credenciais.
- **false** – Use o IMDS para obter credenciais.

**ec2\_metadata\_v1\_disabled**- configuração de AWS **config** arquivo compartilhado, **AWS\_EC2\_METADATA\_V1\_DISABLED**: variável de ambiente, **aws.disableEc2MetadataV1**- Propriedade do sistema JVM: somente Java/Kotlin

Se o serviço de metadados de instância versão 1 (IMDSv1) deve ser usado como alternativa se o IMDSv2 falhar.

 Note

Os novos SDKs não são compatíveis com IMDSv1 e, portanto, não oferecem suporte a essa configuração. Para obter detalhes, consulte a tabela [Compatibilidade com AWS SDKs](#).

Valor padrão: `false`.

Valores válidos:

- **true** – Não use o IMDSv1 como alternativa.
- **false** – Use o IMDSv1 como alternativa.

**ec2\_metadata\_service\_endpoint**- configuração de AWS **config** arquivo compartilhado, **AWS\_EC2\_METADATA\_SERVICE\_ENDPOINT**: variável de ambiente, **aws.ec2MetadataServiceEndpoint**- Propriedade do sistema JVM: somente Java/Kotlin

O endpoint de IMDS.

Valor padrão: se `ec2_metadata_service_endpoint_mode` for igual a IPv4, o endpoint padrão será `http://169.254.169.254`. Se `ec2_metadata_service_endpoint_mode` for igual a IPv6, o endpoint padrão será `http://[fd00:ec2::254]`.

Valores válidos: URL válido.

**ec2\_metadata\_service\_endpoint\_mode**- configuração de AWS **config** arquivo compartilhado, **AWS\_EC2\_METADATA\_SERVICE\_ENDPOINT\_MODE**: variável de ambiente, **aws.ec2MetadataServiceEndpointMode**- Propriedade do sistema JVM: somente Java/Kotlin

O modo de endpoint do IMDS.

Valor padrão: IPv4.

Valores válidos: IPv4, IPv6.

#### Note

O provedor de credenciais do IMDS faz parte do [Cadeia de fornecedores de credenciais](#). No entanto, o provedor de credenciais do IMDS só é verificado após vários outros provedores que estão nesta série. Portanto, se você quiser que seu programa use as credenciais desse provedor, você deve remover outros provedores de credenciais válidos da sua configuração ou usar um perfil diferente. Como alternativa, em vez de confiar na cadeia de provedores de credenciais para descobrir automaticamente qual provedor retorna credenciais válidas, especifique o uso do provedor de credenciais IMDS no código. Você pode especificar fontes de credenciais diretamente ao criar clientes de serviço.

## Segurança para credenciais do IMDS

Por padrão, quando o AWS SDK não está configurado com credenciais válidas, o SDK tentará usar o Amazon EC2 Instance Metadata Service (IMDS) para recuperar as credenciais de uma função. AWS Esse comportamento pode ser desativado definindo a variável de ambiente **AWS\_EC2\_METADATA\_DISABLED** como `true`. Isso evita atividades de rede desnecessárias e aumenta a segurança em redes não confiáveis nas quais o serviço de metadados de instância do Amazon EC2 pode ser representado.

#### Note

AWS Clientes SDK configurados com credenciais válidas nunca usarão o IMDS para recuperar credenciais, independentemente de qualquer uma dessas configurações.

## Desabilitando o uso das credenciais do Amazon EC2 IMDS

A forma como você define essa variável de ambiente depende do sistema operacional em uso, bem como se você deseja ou não que a alteração seja persistente.

### Linux e macOS

Os clientes que usam Linux ou macOS podem definir essa variável de ambiente com o comando a seguir:

```
$ export AWS_EC2_METADATA_DISABLED=true
```

Se você quiser que essa configuração seja persistente em várias sessões de shell e reinicializações do sistema, você pode adicionar o comando acima ao seu arquivo de perfil de shell, como `.bash_profile`, `.zsh_profile` ou `.profile`.

### Windows

Os clientes que usam Windows podem definir essa variável de ambiente com o comando a seguir:

```
$ set AWS_EC2_METADATA_DISABLED=true
```

Se você quiser que essa configuração seja persistente em várias sessões de shell e reinicializações do sistema, use o seguinte comando em vez disso:

```
$ setx AWS_EC2_METADATA_DISABLED=true
```

#### Note

O comando `setx` não aplica o valor à sessão atual do shell, então você precisará recarregar ou reabrir o shell para que a alteração entre em vigor.

## Compatibilidade com AWS SDKs

Os SDKs a seguir são compatíveis com os recursos e configurações descritos neste tópico. Quaisquer exceções parciais estão anotadas. Todas as configurações de propriedade do sistema JVM são suportadas pelo AWS SDK for Java e pelo AWS SDK para Kotlin único.

SDK	Compartilhado	Notas ou mais informações
<a href="#">AWS CLI v2</a>	Sim	
<a href="#">SDK para C++</a>	Sim	
<a href="#">SDK para Go V2 (1.x)</a>	Sim	
<a href="#">SDK para Go 1.x (V1)</a>	Sim	Para usar as configurações do arquivo config compartilhado, você deve ativar o carregamento do arquivo de configuração; consulte <a href="#">Sessões</a> .
<a href="#">SDK para Java 2.x</a>	Sim	
<a href="#">SDK para Java 1.x</a>	Parcial	Propriedades do sistema JVM: Use <code>com.amazonaws.sdk.disableEc2MetadataV1</code> em vez de <code>aws.disableEc2MetadataV1</code> ; <code>aws.ec2MetadataServiceEndpoint</code> e <code>aws.ec2MetadataServiceEndpointMode</code> não suportado.
<a href="#">SDK para 3.x JavaScript</a>	Sim	
<a href="#">SDK para 2.x JavaScript</a>	Sim	
<a href="#">SDK para Kotlin</a>	Sim	Não usa o fallback do IMDSv1.
<a href="#">SDK para .NET 3.x</a>	Sim	
<a href="#">SDK para PHP 3.x</a>	Sim	
<a href="#">SDK para Python (Boto3)</a>	Sim	
<a href="#">SDK para Ruby 3.x</a>	Sim	
<a href="#">SDK para Rust</a>	Sim	Não usa o fallback do IMDSv1.
<a href="#">Ferramentas para PowerShell</a>	Sim	Você pode desativar o fallback do IMDSv1 explicitamente no código usando. <code>[Amazon.Util.EC2InstanceMetadata]::EC2MetadataV1Disabled = \$true</code>

## Provedor de credenciais de processo

Os SDKs fornecem uma forma de estender a cadeia de fornecedores de credenciais para casos de uso personalizados.

O IAM Roles Anywhere fornece uma maneira de obter credenciais temporárias para uma carga de trabalho ou processo executado fora do. AWS Para configurar `credential_process` para este uso, consulte [IAM Roles Anywhere](#).

### Warning

A seguir, a descrição de um método de obtenção de credenciais de um processo externo. Isto pode ser potencialmente perigoso, portanto, prossiga com cuidado. Outros provedores de credenciais devem ser preferidos, se possível. Ao usar esta opção, certifique-se de que o `config` arquivo esteja o mais bloqueado possível usando as melhores práticas de segurança para seu sistema operacional. Verifique se sua ferramenta de credenciais personalizada não grava informações secretas em `StdErr` porque os SDKs e a AWS CLI podem capturar e registrar essas informações, possivelmente expondo-as a usuários não autorizados.

Configure esta funcionalidade usando o seguinte:

### **credential\_process**- configuração de AWS **config** arquivo compartilhado

Especifica um comando externo que o SDK ou uma ferramenta executa em seu nome para gerar ou recuperar credenciais de autenticação a serem usadas. A configuração especifica o nome de um programa/comando que o SDK invocará. Quando o SDK invoca o processo, ele espera que o processo grave dados JSON em `stdout`. O provedor personalizado deve retornar informações em um formato específico. Essas informações contêm as credenciais que o SDK ou a ferramenta podem usar para autenticar você.

### Note

O provedor de credenciais do processo faz parte do [Cadeia de fornecedores de credenciais](#). No entanto, o provedor de credenciais do processo só é verificado após vários outros provedores que estão nesta série. Portanto, se você quiser que seu programa use as credenciais deste provedor, você deve remover outros provedores de credenciais válidos da

sua configuração ou usar um perfil diferente. Como alternativa, em vez de confiar na cadeia de fornecedores de credenciais para descobrir automaticamente qual provedor retorna credenciais válidas, especifique o uso do provedor de credenciais do processo no código. Você pode especificar fontes de credenciais diretamente ao criar clientes de serviço.

## Especificando o caminho para o programa de credenciais

O valor da configuração é uma string que contém um caminho para um programa que o SDK ou a ferramenta de desenvolvimento executa em seu nome:

- O caminho e o nome do arquivo podem consistir somente dos seguintes caracteres: A-Z, a-z, 0-9, hífen (-), sublinhado (\_), barra (/), barra invertida (\) e espaço.
- Se o caminho ou o nome do arquivo contiver um espaço, coloque o caminho completo e o nome do arquivo entre aspas duplas (" ").
- Se um nome de parâmetro ou um valor de parâmetro tiver um espaço, coloque esse elemento entre aspas duplas (" "). Coloque somente o nome ou o valor entre aspas, não o par.
- Não inclua variáveis de ambiente nas strings. Por exemplo, não inclua \$HOME ou %USERPROFILE %.
- Não especifique a pasta base como ~. \* Você deve especificar o caminho completo ou o nome do arquivo base. Se houver um nome de arquivo base, o sistema tentará encontrar o programa nas pastas especificadas pela variável de PATH ambiente.

O exemplo a seguir mostra a configuração de `credential_process` no `config` arquivo compartilhado no Linux/macOS.

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

O exemplo a seguir mostra a configuração de `credential_process` no `config` arquivo compartilhado no Windows.

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



## Saída válida do programa de credenciais

O SDK executa o comando conforme especificado no perfil e em seguida lê os dados do fluxo de saída padrão. O comando especificado, seja um script ou um programa binário, deverá gerar a saída JSON em STDOUT que corresponde à sintaxe a seguir.

```
{
  "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

No momento da elaboração deste documento, a chave `Version` deve ser definida como 1. Isso pode aumentar ao longo do tempo conforme a estrutura evolui.

A chave `Expiration` é um timestamp no formato RFC3339. Se a chave `Expiration` não estiver presente na saída da ferramenta, o SDK vai supor que as credenciais são de longo prazo que não são atualizadas. Caso contrário, as credenciais serão consideradas temporárias e serão atualizadas automaticamente com a nova execução do comando `credential_process` antes de expirarem.

### Note

O SDK não armazena em cache as credenciais do processo externo como faz com credenciais `assume-role`. Se o armazenamento em cache for obrigatório, implemente-o no processo externo.

O processo externo pode retornar um código de retorno diferente de zero para indicar que ocorreu um erro ao recuperar as credenciais.

## Compatibilidade com AWS SDKs

Os SDKs a seguir são compatíveis com os recursos e configurações descritos neste tópico. Quaisquer exceções parciais estão anotadas. Todas as configurações de propriedade do sistema JVM são suportadas pelo AWS SDK for Java e pelo AWS SDK para Kotlin único.

SDK	Compartilhado	Notas ou mais informações
<a href="#">AWS CLI v2</a>	Sim	
<a href="#">SDK para C++</a>	Sim	
<a href="#">SDK for Go V2 (1.x)</a>	Sim	
<a href="#">SDK for Go 1.x (V1)</a>	Sim	Para usar as configurações do arquivo config compartilhado, você deve ativar o carregamento do arquivo de configuração; consulte <a href="#">Sessões</a> .
<a href="#">SDK para Java 2.x</a>	Sim	
<a href="#">SDK para Java 1.x</a>	Sim	
<a href="#">SDK para 3.x JavaScript</a>	Sim	
<a href="#">SDK para 2.x JavaScript</a>	Sim	
<a href="#">SDK for Kotlin</a>	Sim	
<a href="#">SDK for .NET 3.x</a>	Sim	
<a href="#">SDK for PHP 3.x</a>	Sim	
<a href="#">SDK for Python (Boto3)</a>	Sim	
<a href="#">SDK for Ruby 3.x</a>	Sim	
<a href="#">SDK for Rust</a>	Sim	
<a href="#">Ferramentas para PowerShell</a>	Sim	

## AWS Recursos padronizados de SDKs e ferramentas

Muitos atributos foram padronizados para padrões consistentes e para funcionar da mesma forma em muitos SDKs. Essa consistência aumenta a produtividade e a clareza ao codificar em vários

SDKs. Todas as configurações podem ser substituídas no código. Consulte sua API específica do SDK para obter detalhes.

### Important

Nem todos os SDKs oferecem suporte a todos os atributos, ou mesmo a todos os aspectos de um atributo.

## Tópicos

- [ID da aplicação](#)
- [Metadados da instância do Amazon EC2](#)
- [Pontos de acesso Amazon S3](#)
- [Pontos de acesso de várias regiões do Amazon S3](#)
- [Região da AWS](#)
- [AWS STS Endpoints regionalizados](#)
- [Endpoints de pilha dupla e FIPS](#)
- [Descoberta de endpoint](#)
- [Definições gerais da configuração](#)
- [Cliente de IMDS](#)
- [Comportamento de repetição](#)
- [Compactação de solicitações](#)
- [Endpoints específicos de serviço](#)
- [Padrões de configuração inteligente](#)

## ID da aplicação

Um único Conta da AWS pode ser usado por vários aplicativos de clientes para fazer chamadas para Serviços da AWS. O ID do aplicativo fornece uma maneira de os clientes identificarem qual aplicativo de origem fez um conjunto de chamadas usando um Conta da AWS. AWS Os SDKs e os serviços não usam nem interpretam esse valor a não ser para trazê-lo de volta às comunicações com os clientes. Por exemplo, esse valor pode ser incluído em e-mails operacionais ou no AWS

Health Dashboard para identificar com exclusividade quais dos seus aplicativos estão associados à notificação.

Configure esta funcionalidade usando o seguinte:

**sdk\_ua\_app\_id**- configuração de AWS **config** arquivo compartilhado, **AWS\_SDK\_UA\_APP\_ID** - variável de ambiente, **aws.userAgentAppId**- Propriedade do sistema JVM: somente Java/Kotlin

Essa configuração é uma string exclusiva que você atribui ao seu aplicativo para identificar para quais aplicativos em um determinado aplicativo Conta da AWS fazem chamadas AWS.

Valor padrão: None

Valores válidos: string com comprimento máximo de 50. Letras, números e os seguintes caracteres especiais são permitidos: !\$,%,&,\*,+,-,.,,^,\_,`,`|,~.

Exemplo de configuração desse valor no arquivo config:

```
[default]
sdk_ua_app_id=ABCDEF
```

Exemplo de configuração de variáveis de ambiente no Linux/macOS por meio da linha de comando:

```
export AWS_SDK_UA_APP_ID=ABCDEF
export AWS_SDK_UA_APP_ID="ABC DEF"
```

Exemplo do Windows de configuração de variáveis de ambiente por meio da linha de comando:

```
setx AWS_SDK_UA_APP_ID ABCDEF
setx AWS_SDK_UA_APP_ID="ABC DEF"
```

Se você incluir símbolos que tenham um significado especial para a concha que está sendo usada, escape do valor conforme apropriado.

## Compatibilidade com AWS SDKs

Os SDKs a seguir são compatíveis com os recursos e configurações descritos neste tópico. Quaisquer exceções parciais estão anotadas. Todas as configurações de propriedade do sistema JVM são suportadas pelo AWS SDK for Java e pelo AWS SDK para Kotlin único.

SDK	Compartilhado	Notas ou mais informações
<a href="#">AWS CLI v2</a>	Não	
<a href="#">SDK for C++</a>	Sim	arquivo compartilhado config não suportado.
<a href="#">SDK para Go V2 (1.x)</a>	Sim	
<a href="#">SDK for Go 1.x (V1)</a>	Não	
<a href="#">SDK para Java 2.x</a>	Parcial	Configuração de config arquivo compartilhado não suportada; variável de ambiente não suportada.
<a href="#">SDK para Java 1.x</a>	Não	
<a href="#">SDK para 3.x JavaScript</a>	Sim	
<a href="#">SDK para 2.x JavaScript</a>	Não	
<a href="#">SDK for Kotlin</a>	Sim	
<a href="#">SDK for .NET 3.x</a>	Sim	As variáveis de ambiente não são compatíveis.
<a href="#">SDK para PHP 3.x</a>	Sim	
<a href="#">SDK for Python (Boto3)</a>	Sim	
<a href="#">SDK for Ruby 3.x</a>	Sim	
<a href="#">SDK for Rust</a>	Sim	
<a href="#">Ferramentas para PowerShell</a>	Não	

## Metadados da instância do Amazon EC2

O Amazon EC2 fornece um serviço em instâncias chamado serviço de metadados de instância (IMDS). Para saber mais sobre esse serviço, consulte [Metadados da instância e dados do usuário](#) no Guia do usuário do Amazon EC2 [ou Metadados da instância e dados do usuário no Guia do](#)

[usuário do](#) Amazon EC2. Ao tentar recuperar credenciais em uma instância do Amazon EC2 que foi configurada com um perfil do IAM, uma conexão ao serviço de metadados da instância é ajustável.

Configure essa funcionalidade usando o seguinte:

**metadata\_service\_num\_attempts**- configuração de AWS **config** arquivo compartilhado,  
**AWS\_METADATA\_SERVICE\_NUM\_ATTEMPTS**: variável de ambiente

Esta configuração especifica o número de tentativas totais a serem feitas antes de desistir ao recuperar dados do serviço de metadados de instância.

Valor padrão: 1

Valores válidos: número maior ou igual a 1.

**metadata\_service\_timeout**- configuração de AWS **config** arquivo compartilhado,  
**AWS\_METADATA\_SERVICE\_TIMEOUT**: variável de ambiente

Especifica o número de segundos antes de atingir o tempo limite ao recuperar dados do serviço de metadados da instância.

Valor padrão: 1

Valores válidos: número maior ou igual a 1.

Exemplo de configuração desses valores no arquivo config:

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

Exemplo de configuração de variáveis de ambiente para Linux/macOS por meio da linha de comando:

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

Exemplo do Windows de configuração de variáveis de ambiente por meio da linha de comando:

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

## Compatibilidade com AWS SDKs

Os SDKs a seguir são compatíveis com os recursos e configurações descritos neste tópico. Quaisquer exceções parciais estão anotadas. Todas as configurações de propriedade do sistema JVM são suportadas pelo AWS SDK for Java e pelo AWS SDK para Kotlin único.

SDK	Compatível	Notas ou mais informações
<a href="#">AWS CLI v2</a>	Sim	
<a href="#">SDK para C++</a>	Não	
<a href="#">SDK para Go V2 (1.x)</a>	Não	
<a href="#">SDK para Go 1.x (V1)</a>	Não	
<a href="#">SDK para Java 2.x</a>	Não	
<a href="#">SDK para Java 1.x</a>	Parcial	Somente <code>AWS_METADATA_SERVICE_TIMEOUT</code> é suportado.
<a href="#">SDK para 3.x JavaScript</a>	Não	
<a href="#">SDK para 2.x JavaScript</a>	Não	
<a href="#">SDK para Kotlin</a>	Não	
<a href="#">SDK para .NET 3.x</a>	Não	
<a href="#">SDK para PHP 3.x</a>	Sim	
<a href="#">SDK para Python (Boto3)</a>	Sim	
<a href="#">SDK para Ruby 3.x</a>	Não	
<a href="#">SDK para Rust</a>	Não	
<a href="#">Ferramentas para PowerShell</a>	Não	

## Pontos de acesso Amazon S3

O serviço Amazon S3 fornece pontos de acesso como uma forma alternativa de interagir com os buckets do Amazon S3. Os pontos de acesso têm políticas e configurações exclusivas aplicadas a eles, em vez de diretamente ao bucket. Com AWS os SDKs, você pode usar o ponto de acesso Amazon Resource Names (ARNs) no campo de bucket para operações de API em vez de especificar o nome do bucket explicitamente. Eles são usados para operações específicas, como usar um ponto de acesso ARN com o [GetObject](#) para buscar um objeto de um bucket ou usar um ponto de acesso ARN com o [PutObject](#) para adicionar um objeto a um bucket.

Para saber mais sobre os pontos de acesso Amazon S3 e ARNs, consulte [Uso de pontos de acesso](#) no Guia do usuário do Amazon S3.

Configure esta funcionalidade usando o seguinte:

**s3\_use\_arn\_region**- configuração de AWS **config** arquivo compartilhado, **AWS\_S3\_USE\_ARN\_REGION** - variável de ambiente, **aws.s3UseArnRegion**- Propriedade do sistema JVM: somente Java/Kotlin, Para configurar o valor diretamente no código, consulte diretamente seu SDK específico.

Essa configuração controla se o SDK usa o Região da AWS ARN do ponto de acesso para construir o endpoint regional para a solicitação. O SDK valida que o ARN Região da AWS é servido pela mesma AWS partição configurada pelo cliente Região da AWS para evitar chamadas entre partições que provavelmente falharão. Se definido por multiplicação, a configuração configurada pelo código terá precedência, seguida pela configuração da variável de ambiente.

Valor padrão: `false`

Valores válidos:

- **true**— O SDK usa os ARNs Região da AWS ao construir o endpoint em vez do configurado pelo cliente. Região da AWS Exceção: se a configuração do cliente Região da AWS for FIPS Região da AWS, ela deverá corresponder aos ARNs. Região da AWS Caso contrário, ocorrerá um erro.
- **false** – O SDK usa o Região da AWS configurado pelo cliente ao construir o endpoint.



## Compatibilidade com AWS SDKs

Os SDKs a seguir são compatíveis com os recursos e configurações descritos neste tópico. Quaisquer exceções parciais estão anotadas. Todas as configurações de propriedade do sistema JVM são suportadas pelo AWS SDK for Java e pelo AWS SDK para Kotlin único.

SDK	Compatível	Notas ou mais informações
<a href="#">AWS CLI v2</a>	Sim	
<a href="#">SDK para C++</a>	Sim	
<a href="#">SDK for Go V2 (1.x)</a>	Sim	
<a href="#">SDK for Go 1.x (V1)</a>	Sim	Para usar as configurações do arquivo config compartilhado, você deve ativar o carregamento do arquivo de configuração; consulte <a href="#">Sessões</a> .
<a href="#">SDK para Java 2.x</a>	Sim	
<a href="#">SDK para Java 1.x</a>	Sim	A propriedade do sistema JVM não é suportada.
<a href="#">SDK para 3.x JavaScript</a>	Sim	
<a href="#">SDK para 2.x JavaScript</a>	Sim	
<a href="#">SDK for Kotlin</a>	Sim	
<a href="#">SDK for .NET 3.x</a>	Sim	Não segue a precedência padrão; o valor do arquivo config compartilhado tem precedência sobre a variável de ambiente.
<a href="#">SDK for PHP 3.x</a>	Sim	
<a href="#">SDK for Python (Boto3)</a>	Sim	
<a href="#">SDK for Ruby 3.x</a>	Sim	
<a href="#">SDK for Rust</a>	Não	

SDK	C: Notas ou mais informações I
<a href="#">Ferramentas para PowerShell</a> I	Sim Não segue a precedência padrão; o valor do arquivo <code>config</code> compartilhado tem precedência sobre a variável de ambiente.

## Pontos de acesso de várias regiões do Amazon S3

Os pontos de acesso multirregionais do Amazon S3 fornecem um endpoint global que as aplicações podem usar para atender a solicitações de buckets do S3 localizados em várias Regiões da AWS. Você pode usar pontos de acesso multirregionais para criar aplicações de várias regiões com a mesma arquitetura usada em uma única região e, em seguida, executar essas aplicações em qualquer lugar do mundo.

Para saber mais sobre pontos de acesso de várias regiões, consulte [Pontos de acesso de várias regiões no Amazon S3](#), no Guia do usuário do Amazon S3.

Para saber mais sobre como os nomes do recurso da Amazon (ARN) de pontos de acesso de várias regiões funcionam, consulte [Fazer solicitações utilizando um ponto de acesso de várias regiões](#) no Guia do usuário do Amazon S3.

Para saber mais sobre pontos de acesso de várias regiões, consulte [Gerenciando Pontos de acesso de várias regiões](#) no Guia do usuário do Amazon S3.

O algoritmo SigV4a é a implementação de assinatura usada para assinar as solicitações globais da região. Este algoritmo é obtido pelo SDK por meio de uma dependência em [Bibliotecas do Common Runtime \(CRT\) da AWS](#).

Configure esta funcionalidade usando o seguinte:

**s3\_disable\_multiregion\_access\_points**- configuração de AWS `config` arquivo compartilhado, **AWS\_S3\_DISABLE\_MULTIREGION\_ACCESS\_POINTS** - variável de ambiente, **aws.s3DisableMultiRegionAccessPoints**- Propriedade do sistema JVM: somente Java/Kotlin, Para configurar o valor diretamente no código, consulte diretamente seu SDK específico.

Esta configuração controla se o SDK pode tentar solicitações entre regiões. Se definido por multiplicação, a configuração configurada pelo código terá precedência, seguida pela configuração da variável de ambiente.

Valor padrão: `false`

Valores válidos:

- **true** — Interrompe o uso de solicitações entre regiões.
- **false** — Permite solicitações entre regiões usando pontos de acesso multirregionais.

## Compatibilidade com AWS SDKs

Os SDKs a seguir são compatíveis com os recursos e configurações descritos neste tópico. Quaisquer exceções parciais estão anotadas. Todas as configurações de propriedade do sistema JVM são suportadas pelo AWS SDK for Java e pelo AWS SDK para Kotlin único.

SDK	Compatível	Notas ou mais informações
<a href="#">AWS CLI v2</a>	Sim	
<a href="#">SDK para C++</a>	Sim	
<a href="#">SDK for Go V2 (1.x)</a>	Sim	
<a href="#">SDK for Go 1.x (V1)</a>	Não	
<a href="#">SDK para Java 2.x</a>	Sim	
<a href="#">SDK para Java 1.x</a>	Não	
<a href="#">SDK para 3.x JavaScript</a>	Sim	
<a href="#">SDK para 2.x JavaScript</a>	Não	
<a href="#">SDK for Kotlin</a>	Sim	
<a href="#">SDK for .NET 3.x</a>	Sim	
<a href="#">SDK for PHP 3.x</a>	Sim	
<a href="#">SDK for Python (Boto3)</a>	Sim	
<a href="#">SDK for Ruby 3.x</a>	Sim	

SDK	C	Notas ou mais informações
<a href="#">SDK for Rust</a>	Sim	
<a href="#">Ferramentas para PowerShell</a>	Sim	

## Região da AWS

Regiões da AWS são um conceito importante a ser entendido ao trabalhar com Serviços da AWS.

Com Regiões da AWS, você pode acessar aqueles Serviços da AWS que residem fisicamente em uma área geográfica específica. Isso pode ser útil para manter os seus dados e aplicativos em execução próximo ao lugar em que você e os seus usuários os acessarão. As regiões fornecem tolerância a falhas, estabilidade e resiliência e também podem reduzir a latência. Com Regiões, você pode criar recursos redundantes que permanecem disponíveis e não são afetados por uma interrupção regional.

A maioria das AWS service (Serviço da AWS) solicitações está associada a uma região geográfica específica. Os atributos que você cria em uma Região não existem em qualquer outra Região, a menos que você use explicitamente um atributo de replicação oferecido por um AWS service (Serviço da AWS). Por exemplo, o Amazon S3 e o Amazon EC2 oferecem suporte à replicação entre Regiões. Alguns serviços, como o IAM, não têm Recursos regionais.

O Referência geral da AWS contém as seguintes informações:

- Para entender a relação entre Regiões e endpoints e ver uma lista dos endpoints regionais existentes, consulte [Endpoints de serviço da AWS](#).
- Para ver a lista atual de todas as regiões e endpoints compatíveis com cada uma AWS service (Serviço da AWS), consulte [Endpoints e cotas de serviço](#).

### Criar clientes de serviço

Para acessar programaticamente Serviços da AWS, os SDKs usam uma classe/objeto cliente para cada um. AWS service (Serviço da AWS) Se seu aplicativo precisar acessar o Amazon EC2, por exemplo, seu aplicativo criará um objeto cliente do Amazon EC2 para interagir com esse serviço.

Se nenhuma Região for especificada explicitamente para o cliente, o cliente usará como padrão a Região definida por meio da configuração `region` a seguir. No entanto, a Região ativa de um cliente pode ser definida explicitamente para qualquer objeto de cliente individual. Definir a Região desta maneira tem precedência sobre qualquer configuração global para aquele cliente de serviço particular. A Região alternativa é especificada durante a instanciação desse cliente, específica para seu SDK (consulte seu Guia do SDK específico ou a base de código do seu SDK).

Configure esta funcionalidade usando o seguinte:

**region**- configuração de AWS **config** arquivo compartilhado, **AWS\_REGION** - variável de ambiente,

**aws.region**- Propriedade do sistema JVM: somente Java/Kotlin

Especifica o padrão Região da AWS a ser usado para AWS solicitações. Essa região é usada para solicitações de serviço do SDK que não são fornecidas com uma Região específica para uso.

Valor padrão: Nenhum. Você deve especificar esse valor explicitamente.

Valores válidos:

- Qualquer um dos códigos de Região disponíveis para o serviço escolhido, conforme listado em [Endpoints de serviço da AWS](#) na Referência geral AWS . Por exemplo, o valor `us-east-1` define o endpoint para o Leste dos Região da AWS EUA (Norte da Virgínia).
- `aws-global` especifica o endpoint global para serviços que oferecem suporte a um endpoint global separado, além dos endpoints regionais, como AWS Security Token Service (AWS STS) e Amazon Simple Storage Service (Amazon S3).

Exemplo de configuração desse valor no arquivo `config`:

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

Exemplo de configuração de variáveis de ambiente no Linux/macOS por meio da linha de comando:

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

Exemplo de configuração de variáveis de ambiente no Windows por meio da linha de comando:

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

A maioria dos SDKs tem um objeto de “configuração” que está disponível para definir a região padrão a partir do código do aplicativo. Para obter detalhes, consulte seu guia específico para desenvolvedores do AWS SDK.

## Compatibilidade com AWS SDKs

Os SDKs a seguir são compatíveis com os recursos e configurações descritos neste tópico. Quaisquer exceções parciais estão anotadas. Todas as configurações de propriedade do sistema JVM são suportadas pelo AWS SDK for Java e pelo AWS SDK para Kotlin único.

SDK	C	Notas ou mais informações
<a href="#">AWS CLI v2</a>	Sim	AWS CLI v2 usa qualquer valor em <code>AWS_REGION</code> antes de qualquer valor em <code>AWS_DEFAULT_REGION</code> (ambas as variáveis são verificadas).
<a href="#">AWS CLI v1</a>	Sim	AWS CLI v1 usa uma variável de ambiente nomeada <code>AWS_DEFAULT_REGION</code> para essa finalidade.
<a href="#">SDK para C++</a>	Sim	
<a href="#">SDK for Go V2 (1.x)</a>	Sim	
<a href="#">SDK for Go 1.x (V1)</a>	Sim	Para usar as configurações do arquivo <code>config</code> compartilhado, você deve ativar o carregamento do arquivo de configuração; consulte <a href="#">Sessões</a> .
<a href="#">SDK para Java 2.x</a>	Sim	
<a href="#">SDK para Java 1.x</a>	Sim	
<a href="#">SDK para 3.x JavaScript</a>	Sim	
<a href="#">SDK para 2.x JavaScript</a>	Sim	
<a href="#">SDK for Kotlin</a>	Sim	
<a href="#">SDK for .NET 3.x</a>	Sim	

SDK	Comentários
<a href="#">SDK for PHP 3.x</a>	Sim
<a href="#">SDK for Python (Boto3)</a>	Sim Este SDK usa uma variável de ambiente nomeada <code>AWS_DEFAULT_REGION</code> para essa finalidade.
<a href="#">SDK for Ruby 3.x</a>	Sim
<a href="#">SDK for Rust</a>	Sim
<a href="#">Ferramentas para PowerShell</a>	Sim

## AWS STS Endpoints regionalizados

Por padrão, AWS Security Token Service (AWS STS) está disponível como um serviço global e todas as AWS STS solicitações vão para um único endpoint em `https://sts.amazonaws.com`. As solicitações globais são mapeadas para a região Leste dos EUA (Norte da Virgínia). AWS recomenda o uso de AWS STS endpoints regionais em vez do endpoint global. Para obter mais informações sobre AWS STS endpoints, [Endpoints](#) na Referência da AWS Security Token Service API.

Configure esta funcionalidade usando o seguinte:

**sts\_regional\_endpoints**- configuração de AWS **config** arquivo compartilhado,  
**AWS\_STS\_REGIONAL\_ENDPOINTS** - variável de ambiente

Essa configuração especifica como o SDK ou a ferramenta determina o AWS service (Serviço da AWS) endpoint que ele usa para se comunicar com o AWS Security Token Service (AWS STS).

Valor padrão: `legacy`

### Note

Todas as novas versões principais do SDK lançadas após julho de 2022 terão como padrão `regional`. As novas versões principais do SDK podem remover essa configuração e usar o comportamento `regional`. Para reduzir o impacto futuro em

relação a essa alteração, recomendamos que você comece a usar `regional` em seu aplicativo sempre que possível.

Valores válidos: (Valor recomendado: `regional`)

- **legacy**— Usa o AWS STS endpoint `globalsts.amazonaws.com`, para as seguintes AWS regiões: `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`, e `us-west-2`. Todas as demais Regiões usam automaticamente o respectivo endpoint regional.
- **regional**— O SDK ou a ferramenta sempre usa o AWS STS endpoint da região atualmente configurada. Por exemplo, se o cliente estiver configurado para usar `us-west-2`, todas as chamadas AWS STS serão feitas para o endpoint `regionalsts.us-west-2.amazonaws.com`, em vez do `sts.amazonaws.com` endpoint global. Para enviar uma solicitação para o endpoint global enquanto a configuração é habilitada, você pode definir a Região como `aws-global`.

Exemplo de configuração desses valores no arquivo `config`:

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

Exemplo de configuração de variáveis de ambiente para Linux/macOS por meio da linha de comando:

```
export AWS_STS_REGIONAL_ENDPOINTS=regional
```

Exemplo do Windows de configuração de variáveis de ambiente por meio da linha de comando:

```
setx AWS_STS_REGIONAL_ENDPOINTS regional
```

## Compatibilidade com AWS SDKs

Os SDKs a seguir são compatíveis com os recursos e configurações descritos neste tópico. Quaisquer exceções parciais estão anotadas. Todas as configurações de propriedade do sistema JVM são suportadas pelo AWS SDK for Java e pelo AWS SDK para Kotlin único.



SDK	Compartilhado	Notas ou mais informações
<a href="#">AWS CLI v2</a>	Parcial	O valor padrão é regional.
<a href="#">SDK for C++</a>	Parcial	A variável de ambiente e a configuração do arquivo config não são suportadas. O SDK funciona com a configuração regional.
<a href="#">SDK for Go V2 (1.x)</a>	Sim	
<a href="#">SDK for Go 1.x (V1)</a>	Sim	Para usar as configurações do arquivo config compartilhado, você deve ativar o carregamento do arquivo de configuração; consulte <a href="#">Sessões</a> .
<a href="#">SDK para Java 2.x</a>	Sim	
<a href="#">SDK para Java 1.x</a>	Sim	
<a href="#">SDK para 3.x JavaScript</a>	Sim	
<a href="#">SDK para 2.x JavaScript</a>	Sim	
<a href="#">SDK for Kotlin</a>	Não	
<a href="#">SDK for .NET 3.x</a>	Sim	
<a href="#">SDK for PHP 3.x</a>	Sim	
<a href="#">SDK for Python (Boto3)</a>	Sim	
<a href="#">SDK for Ruby 3.x</a>	Sim	
<a href="#">SDK for Rust</a>	Sim	
<a href="#">Ferramentas para PowerShell</a>	Sim	

## Endpoints de pilha dupla e FIPS

Configure esta funcionalidade usando o seguinte:

**use\_dualstack\_endpoint**- configuração de AWS **config** arquivo compartilhado,  
**AWS\_USE\_DUALSTACK\_ENDPOINT** - variável de ambiente, **aws.useDualstackEndpoint**-  
Propriedade do sistema JVM: somente Java/Kotlin

Ativa ou desativa se o SDK enviará solicitações para endpoints de pilha dupla. Para saber mais sobre endpoints de pilha dupla, que suportam tráfego IPv4 e IPv6, consulte [Como usar endpoints de pilha dupla do Amazon S3](#) no Guia do usuário do Amazon Simple Storage Service. Endpoints de pilha dupla estão disponíveis para alguns serviços em algumas regiões.

Valor padrão: `false`

Valores válidos:

- **true** – O SDK ou a ferramenta tentará usar endpoints de pilha dupla para fazer solicitações de rede. Se não existir um endpoint de pilha dupla para o serviço e/ou Região da AWS, a solicitação falhará.
- **false** – O SDK ou a ferramenta não usarão endpoints de pilha dupla para fazer solicitações de rede.

**use\_fips\_endpoint**- configuração de AWS **config** arquivo compartilhado,  
**AWS\_USE\_FIPS\_ENDPOINT** - variável de ambiente, **aws.useFipsEndpoint**- Propriedade do sistema JVM: somente Java/Kotlin

Ativa ou desativa se o SDK enviará solicitações para endpoints de pilha dupla. Os Federal Information Processing Standards (FIPS) são um conjunto de requisitos de segurança do governo dos EUA para dados e sua criptografia. Agências governamentais, parceiros e aqueles que desejam fazer negócios com o governo federal devem seguir as diretrizes do FIPS. Diferentemente dos AWS endpoints padrão, os endpoints FIPS usam uma biblioteca de software TLS compatível com o FIPS 140-2. Se essa configuração estiver ativada e não existir um endpoint FIPS para o serviço em seu Região da AWS, a AWS chamada poderá falhar. [Endpoints específicos de serviço](#) e a `--endpoint-url` opção de AWS Command Line Interface substituir essa configuração.

Para saber mais sobre outras formas de especificar endpoints FIPS por Região da AWS, consulte [FIPS Endpoints](#) por serviço. Para obter mais informações sobre os endpoints do serviço Amazon Elastic Compute Cloud, consulte [Endpoints de pilha dupla \(IPv4 e IPv6\)](#) na Amazon EC2 API Reference.

Valor padrão: `false`

Valores válidos:

- **true** – O SDK ou a ferramenta enviará solicitações para endpoints compatíveis com FIPS.
- **false** – O SDK ou a ferramenta não enviará solicitações para endpoints compatíveis com FIPS.

## Compatibilidade com AWS SDKs

Os SDKs a seguir são compatíveis com os recursos e configurações descritos neste tópico. Quaisquer exceções parciais estão anotadas. Todas as configurações de propriedade do sistema JVM são suportadas pelo AWS SDK for Java e pelo AWS SDK para Kotlin único.

SDK	Compatível	Notas ou mais informações
<a href="#">AWS CLI v2</a>	Sim	
<a href="#">SDK para C++</a>	Sim	
<a href="#">SDK for Go V2 (1.x)</a>	Sim	
<a href="#">SDK for Go 1.x (V1)</a>	Sim	Para usar as configurações do arquivo <code>config</code> compartilhado, você deve ativar o carregamento do arquivo de configuração; consulte <a href="#">Sessões</a> .
<a href="#">SDK para Java 2.x</a>	Sim	
<a href="#">SDK para Java 1.x</a>	Não	
<a href="#">SDK para 3.x JavaScript</a>	Sim	
<a href="#">SDK para 2.x JavaScript</a>	Sim	
<a href="#">SDK for Kotlin</a>	Sim	
<a href="#">SDK for .NET 3.x</a>	Sim	
<a href="#">SDK for PHP 3.x</a>	Sim	

SDK	C	Notas ou mais informações
<a href="#">SDK for Python (Boto3)</a>	Sim	
<a href="#">SDK for Ruby 3.x</a>	Sim	
<a href="#">SDK for Rust</a>	Sim	
<a href="#">Ferramentas para PowerShell</a>	Sim	

## Descoberta de endpoint

Os SDKs usam a descoberta de endpoints para acessar os endpoints de serviço (URLs para acessar vários recursos), mantendo a flexibilidade de alterar os URLs AWS conforme necessário. Dessa forma, seu código pode detectar automaticamente novos endpoints. Não há endpoints fixos para alguns serviços. Em vez disso, você obtém os endpoints disponíveis durante o runtime fazendo uma solicitação para obter os endpoints primeiro. Depois de recuperar os endpoints disponíveis, o código usa o endpoint para acessar outras operações. Por exemplo, para o Amazon Timestream, o SDK faz uma solicitação `DescribeEndpoints` para recuperar os endpoints disponíveis e, em seguida, usa esses endpoints para concluir operações específicas, como `CreateDatabase` ou `CreateTable`.

Configure esta funcionalidade usando o seguinte:

**endpoint\_discovery\_enabled**- configuração de AWS **config** arquivo compartilhado, **AWS\_ENABLE\_ENDPOINT\_DISCOVERY** - variável de ambiente, **aws.endpointDiscoveryEnabled**- Propriedade do sistema JVM: somente Java/Kotlin, Para configurar o valor diretamente no código, consulte diretamente seu SDK específico.

Ativa ou desativa a descoberta de endpoints para o DynamoDB.

A descoberta de endpoints é necessária no Timestream e opcional no Amazon DynamoDB. Essa configuração é padronizada `true` ou `false` depende de o serviço exigir a descoberta do endpoint. As solicitações de Timestream são padronizadas para `true`, e as solicitações do Amazon DynamoDB, como padrão. `false`

Valores válidos:

- **true** – O SDK deve tentar descobrir automaticamente um endpoint para serviços em que a descoberta de endpoint é opcional.
- **false** – O SDK deve tentar descobrir automaticamente um endpoint para serviços em que a descoberta de endpoint é opcional.

## Compatibilidade com AWS SDKs

Os SDKs a seguir são compatíveis com os recursos e configurações descritos neste tópico. Quaisquer exceções parciais estão anotadas. Todas as configurações de propriedade do sistema JVM são suportadas pelo AWS SDK for Java e pelo AWS SDK para Kotlin único.

SDK	Compatibilidade	Notas ou mais informações
<a href="#">AWS CLI v2</a>	Sim	
<a href="#">SDK para C++</a>	Sim	
<a href="#">SDK for Go V2 (1.x)</a>	Sim	
<a href="#">SDK for Go 1.x (V1)</a>	Sim	Para usar as configurações do arquivo config compartilhado, você deve ativar o carregamento do arquivo de configuração; consulte <a href="#">Sessões</a> .
<a href="#">SDK para Java 2.x</a>	Sim	O SDK for Java 2.x <code>AWS_ENDPOINT_DISCOVERY_ENABLED</code> usa o nome da variável de ambiente.
<a href="#">SDK para Java 1.x</a>	Parcial	A propriedade do sistema JVM não é suportada.
<a href="#">SDK para 3.x JavaScript</a>	Sim	
<a href="#">SDK para 2.x JavaScript</a>	Sim	
<a href="#">SDK for Kotlin</a>	Sim	
<a href="#">SDK for .NET 3.x</a>	Sim	
<a href="#">SDK for PHP 3.x</a>	Sim	

SDK	Compatível	Notas ou mais informações
<a href="#">SDK for Python (Boto3)</a>	Sim	
<a href="#">SDK for Ruby 3.x</a>	Sim	
<a href="#">SDK for Rust</a>	Parcial	Compatível somente com Timestream.
<a href="#">Ferramentas para PowerShell</a>	Sim	

## Definições gerais da configuração

Os SDKs oferecem suporte a algumas configurações gerais que definem os comportamentos gerais do SDK.

Configure esta funcionalidade usando o seguinte:

**api\_versions**- configuração de AWS **config** arquivo compartilhado

Alguns AWS serviços mantêm várias versões de API para oferecer suporte à compatibilidade com versões anteriores. Por padrão, as operações do SDK e AWS CLI usam a versão de API mais recente disponível. Para exigir que uma versão específica da API seja usada em suas solicitações, inclua a configuração `api_versions` em seu perfil.

Valor padrão: nenhum. (A versão mais recente da API é usada pelo SDK.)

Valores válidos: essa é uma configuração aninhada seguida por uma ou mais linhas recuadas, cada uma identificando um AWS serviço e a versão da API a ser usada. Consulte a documentação do AWS serviço para entender quais versões de API estão disponíveis.

O exemplo define uma versão específica da API para dois AWS serviços no `config` arquivo. Essas versões de API são usadas apenas para comandos que são executados sob o perfil que contém essas configurações. Os comandos para qualquer outro serviço usam a versão mais recente da API desse serviço.

```
api_versions =
    ec2 = 2015-03-01
```

```
cloudfront = 2015-09-017
```

**ca\_bundle**- configuração de AWS **config** arquivo compartilhado, **AWS\_CA\_BUNDLE** - variável de ambiente

O caminho especifica o caminho para um pacote de certificado personalizado (um arquivo com extensão .pem) a ser usado ao estabelecer conexões SSL/TLS.

Valor padrão: nenhum

Valores válidos: especifique o caminho completo ou o nome do arquivo base. Se houver um nome de arquivo base, o sistema tentará encontrar o programa nas pastas especificadas pela variável de ambiente PATH.

Exemplo de configuração desse valor no arquivo config:

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

Exemplo de configuração de variáveis de ambiente no Linux/macOS por meio da linha de comando:

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

Exemplo do Windows de configuração de variáveis de ambiente via linha de comando:

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

**output**- configuração de AWS **config** arquivo compartilhado

Especifica como os resultados são formatados no e em outros AWS SDKs AWS CLI e ferramentas.

Valor padrão: json

Valores válidos:

- **json**: a saída é formatada como uma string [JSON](#).
- **yaml**: a saída é formatada como uma string [YAML](#).
- **yaml-stream**: a saída é transmitida e formatada como uma string [YAML](#). A transmissão possibilita um manuseio mais rápido de tipos de dados grandes.

- **text**: a saída é formatada como várias linhas de valores de string separados por tabulação. Isso pode ser útil para passar a saída para um processador de texto, como `grep`, `sed` ou `awk`.
- **table**: a saída é formatada como uma tabela usando os caracteres `+|-` para formar as bordas da célula. Geralmente, a informação é apresentada em um formato "amigável", que é muito mais fácil de ler do que outros, mas não tão útil programaticamente.

### **parameter\_validation**- configuração de AWS **config** arquivo compartilhado

Especifica se o SDK ou a ferramenta tentará validar os parâmetros da linha de comando antes de enviá-los para o endpoint de serviço da AWS .

Valor padrão: `true`

Valores válidos:

- **true** – O padrão. O SDK ou a ferramenta executa validação de parâmetros da linha de comando no lado do cliente. Isto ajuda o SDK ou a ferramenta a confirmar se os parâmetros são válidos e a detectar alguns erros. O SDK ou a ferramenta podem rejeitar solicitações que não são válidas antes de enviar solicitações para o endpoint do AWS serviço.
- **false**— O SDK ou a ferramenta não valida os parâmetros da linha de comando antes de enviá-los ao endpoint do AWS serviço. O endpoint do AWS serviço é responsável por validar todas as solicitações e rejeitar solicitações que não são válidas.

## Compatibilidade com AWS SDKs

Os SDKs a seguir são compatíveis com os recursos e configurações descritos neste tópico.

Quaisquer exceções parciais estão anotadas. Todas as configurações de propriedade do sistema JVM são suportadas pelo AWS SDK for Java e pelo AWS SDK para Kotlin único.

SDK	Compatibilidade	Notas ou mais informações
<a href="#">AWS CLI v2</a>	Parcial	<code>api_versions</code> incompatível.
<a href="#">SDK for C++</a>	Sim	
<a href="#">SDK for Go V2 (1.x)</a>	Parcial	<code>api_versions</code> e <code>parameter_validation</code> não são compatíveis.



SDK	Compatibilidade	Notas ou mais informações
<a href="#">SDK for Go 1.x (V1)</a>	Parcial	<code>api_versions</code> e <code>parameter_validation</code> não são compatíveis. Para usar as configurações <code>config</code> do arquivo compartilhado, você deve ativar o carregamento do arquivo de configuração; consulte <a href="#">Sessões</a> .
<a href="#">SDK para Java 2.x</a>	Não	
<a href="#">SDK para Java 1.x</a>	Não	
<a href="#">SDK para 3.x JavaScript</a>	Sim	
<a href="#">SDK para 2.x JavaScript</a>	Sim	
<a href="#">SDK for Kotlin</a>	Não	
<a href="#">SDK para .NET 3.x</a>	Não	
<a href="#">SDK para PHP 3.x</a>	Sim	
<a href="#">SDK for Python (Boto3)</a>	Sim	
<a href="#">SDK for Ruby 3.x</a>	Sim	
<a href="#">SDK for Rust</a>	Não	
<a href="#">Ferramentas para PowerShell</a>	Não	

## Cliente de IMDS

Os SDKs implementam um cliente do serviço de metadados de instância versão 2 (IMDSv2) usando solicitações orientadas a sessões. Para obter mais informações sobre o IMDSv2, consulte [Usar o IMDSv2 no](#) Guia do usuário do Amazon EC2 ou Usar o [IMDSv2](#) no Guia do usuário do Amazon EC2. O cliente IMDS é configurável por meio de um objeto de configuração do cliente disponível na base de código do SDK.

Configure essa funcionalidade usando o seguinte:

**retries**: membro do objeto de configuração do cliente

O número de tentativas adicionais para qualquer solicitação com falha.

Valor padrão: 3

Valores válidos: número maior que zero.

**port**: membro do objeto de configuração do cliente

A porta para o endpoint.

Valor padrão: 80

Valores válidos: número.

**token\_ttl**: membro do objeto de configuração do cliente

O TTL do token.

Valor padrão: 21.600 segundos (6 horas, o tempo máximo alocado).

Valores válidos: número.

**endpoint**: membro do objeto de configuração do cliente

O endpoint de IMDS.

Valor padrão: se `endpoint_mode` for igual a IPv4, o endpoint padrão será `http://169.254.169.254`. Se `endpoint_mode` for igual a IPv6, o endpoint padrão será `http://[fd00:ec2::254]`.

Valores válidos: URL válido.

As opções a seguir são compatíveis com a maioria dos SDKs. Consulte sua base de código específica do SDK para obter detalhes.

**endpoint\_mode**: membro do objeto de configuração do cliente

O modo de endpoint do IMDS.

Valor padrão: IPv4

Valores válidos: IPv4, IPv6

**http\_open\_timeout**: membro do objeto de configuração do cliente (o nome pode variar)

O número de segundos a aguardar até que a conexão seja aberta.

Valor padrão: 1 segundo.

Valores válidos: número maior que zero.

**http\_read\_timeout**: membro do objeto de configuração do cliente (o nome pode variar)

O número de segundos para que um bloco de dados seja lido.

Valor padrão: 1 segundo.

Valores válidos: número maior que zero.

**http\_debug\_output**: membro do objeto de configuração do cliente (o nome pode variar)

Define um fluxo de saída para depuração.

Valor padrão: nenhum.

Valores válidos: um fluxo de I/O válido, como STDOUT.

**backoff**: membro do objeto de configuração do cliente (o nome pode variar)

O número de segundos para dormir entre as novas tentativas ou o cliente forneceu a função de desligamento para chamar. Isto substitui a estratégia padrão de recuo exponencial.

Valor padrão: varia de acordo com o SDK.

Valores válidos: variam de acordo com o SDK. Pode ser um valor numérico ou uma chamada para uma função personalizada.

## Compatibilidade com AWS SDKs

Os SDKs a seguir são compatíveis com os recursos e configurações descritos neste tópico.

Quaisquer exceções parciais estão anotadas. Todas as configurações de propriedade do sistema JVM são suportadas pelo AWS SDK for Java e pelo AWS SDK para Kotlin único.

SDK	Compatibilidade	Notas ou mais informações
<a href="#">AWS CLI v2</a>	Sim	

SDK	Comportamento de repetição	Notas ou mais informações
<a href="#">SDK para C++</a>	Não	O IMDSv2 é usado somente internamente. Consulte <a href="#">Provedor de credenciais IMDS</a> .
<a href="#">SDK para Go V2 (1.x)</a>	Sim	
<a href="#">SDK para Go 1.x (V1)</a>	Sim	
<a href="#">SDK para Java 2.x</a>	Sim	
<a href="#">SDK para Java 1.x</a>	Sim	
<a href="#">SDK para 3.x JavaScript</a>	Sim	
<a href="#">SDK para 2.x JavaScript</a>	Sim	
<a href="#">SDK para Kotlin</a>	Sim	
<a href="#">SDK para .NET 3.x</a>	Sim	
<a href="#">SDK para PHP 3.x</a>	Sim	
<a href="#">SDK para Python (Boto3)</a>	Sim	
<a href="#">SDK para Ruby 3.x</a>	Sim	
<a href="#">SDK para Rust</a>	Sim	
<a href="#">Ferramentas para PowerShell</a>	Sim	

## Comportamento de repetição

O comportamento de repetição inclui configurações sobre como os SDKs tentam se recuperar de falhas resultantes de solicitações feitas para Serviços da AWS.

Configure esta funcionalidade usando o seguinte:

**max\_attempts**- configuração de AWS **config** arquivo compartilhado, **AWS\_MAX\_ATTEMPTS** - variável de ambiente, **aws.maxAttempts**- Propriedade do sistema JVM: somente Java/Kotlin

Especifica o número máximo de tentativas a serem feitas em uma solicitação.

Valor padrão: se esse valor não for especificado, seu padrão dependerá do valor da configuração `retry_mode`:

- Se `retry_mode` for `legacy` – usa um valor padrão específico para seu SDK (consulte o guia de seu SDK específico ou a base de código do seu SDK para ver o `max_attempts` padrão).
- Se `retry_mode` for `standard` – Faz três tentativas.
- Se `retry_mode` for `adaptive` – Faz três tentativas.

Valores válidos: Número maior que zero.

**retry\_mode**- configuração de AWS **config** arquivo compartilhado, **AWS\_RETRY\_MODE** - variável de ambiente, **aws.retryMode**- Propriedade do sistema JVM: somente Java/Kotlin

Especifica como o SDK ou a ferramenta de desenvolvedor tenta novas tentativas.

Valor padrão: `legacy` é a estratégia de repetição padrão.

Valores válidos:

- `legacy` – Específico para seu SDK (consulte seu guia específico do SDK ou a base de código do seu SDK).
- `standard`— O conjunto padrão de regras de repetição em todos os AWS SDKs. Esse modo inclui um conjunto padrão de erros que são repetidos e suporte para cotas de repetição. O número máximo padrão de tentativas com esse modo é três, a menos que `max_attempts` esteja explicitamente configurado.
- `adaptive` – Um modo de repetição experimental que inclui a funcionalidade do modo padrão, mas inclui controle de utilização automática do lado do cliente. Como esse modo é experimental, ele pode mudar o comportamento no futuro.

## Escolher entre os **standard** modos e **adaptive** tentar novamente

Recomendamos que você use o modo de `standard` repetição, a menos que tenha certeza de que seu uso é mais adequado `adaptive`.

**Note**

O `adaptive` modo pressupõe que você esteja agrupando clientes com base no escopo no qual o serviço de back-end pode limitar as solicitações. Se você não fizer isso, as limitações em um recurso podem atrasar as solicitações de um recurso não relacionado se você estiver usando o mesmo cliente para os dois recursos.

Padrão	Adaptável
Casos de uso de aplicativos: todos.	Casos de uso de aplicativos: <ol style="list-style-type: none"> <li>1. Não é sensível à latência.</li> <li>2. O cliente acessa apenas um único recurso, ou você está fornecendo lógica para agrupar seus clientes separadamente pelo recurso de serviço que está sendo acessado.</li> </ol>
Suporta interrupção de circuito para evitar que o SDK tente novamente durante interrupções.	Suporta interrupção de circuito para evitar que o SDK tente novamente durante interrupções.
Usa um recuo exponencial instável em caso de falhas.	Usa durações dinâmicas de recuo para tentar minimizar o número de solicitações com falha, em troca do potencial de maior latência.
Nunca atrasa a primeira tentativa de solicitação, somente as novas tentativas.	Pode acelerar ou atrasar a tentativa de solicitação inicial.

Se você optar por usar o `adaptive` modo, seu aplicativo deverá criar clientes projetados com base em cada recurso que possa ser limitado. Um recurso, nesse caso, é mais refinado do que apenas pensar em cada um. AWS service (Serviço da AWS) Serviços da AWS podem ter dimensões adicionais que eles usam para acelerar as solicitações. Vamos usar o serviço Amazon DynamoDB como exemplo. O DynamoDB Região da AWS usa mais a tabela que está sendo acessada para acelerar as solicitações. Isso significa que uma tabela que seu código está acessando pode ser mais limitada do que outras. Se seu código usou o mesmo cliente para acessar todas as tabelas e as solicitações para uma dessas tabelas forem limitadas, o modo de repetição adaptável reduzirá a taxa de solicitação de todas as tabelas. Seu código deve ser projetado para ter um cliente por região-and-

table par R. Se você tiver uma latência inesperada ao usar o `adaptive` modo, consulte o guia de AWS documentação específico do serviço que você está usando.

## Detalhes da implementação do modo de repetição

A seguir está o pseudocódigo de alto nível para os modos de repetição de repetição `standard` e `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)
  }
}
```

A seguir estão mais detalhes sobre os componentes usados no pseudocódigo:

### **GetSendToken:**

Os buckets de tokens são usados somente no modo de repetição `adaptive`. Os buckets de tokens impõem uma taxa máxima de solicitação exigindo que um token esteja disponível para iniciar uma solicitação. O cliente SDK pode ser configurado para falhar rapidamente na solicitação ou bloquear até que um token fique disponível.

O Limitador da taxa do lado do cliente é um algoritmo que inicialmente permite que as solicitações sejam feitas a qualquer taxa, até o limite de token permitido. No entanto, depois que uma resposta limitada é detectada, o cliente `rate-of-request` fica limitado adequadamente. O subsídio de tokens também é aumentado proporcionalmente se respostas bem-sucedidas forem recebidas.

Com a limitação de taxa adaptativa, os SDKs podem diminuir a taxa na qual as solicitações são enviadas para acomodar melhor a capacidade de. Serviços da AWS

## **SendHTTPRequest:**

A maioria dos AWS SDKs usa uma biblioteca HTTP que usa grupos de conexões para que você possa reutilizar uma conexão existente ao fazer uma solicitação HTTP. Geralmente, as conexões são reutilizadas ao repetir solicitações devido a erros de controle de utilização. As solicitações não são reutilizadas ao tentar novamente devido a erros transitórios.

## **RequestBookkeeping:**

A cota de repetição deverá ser atualizada se a solicitação for bem-sucedida. Somente para o modo de repetição `adaptive`, a variável de estado `maxsendrate` é atualizada com base no tipo de resposta recebida.

## **Retryable:**

Essa etapa determina se uma resposta pode ser repetida com base no seguinte:

- Código de status do HTTP.
- O código de erro retornado do serviço.
- Erros de conexão, definidos como qualquer erro recebido pelo SDK no qual uma resposta HTTP do serviço não é recebida.

Erros transitórios (códigos de status HTTP 400, 408, 500, 502, 503 e 504) e erros de controle de utilização (códigos de status HTTP 400, 403, 429, 502, 503 e 509) podem potencialmente ser repetidos. O comportamento de repetição do SDK é determinado em combinação com códigos de erro ou outros dados do serviço.

## **MAX\_ATTEMPTS:**

Especificado pela configuração do arquivo `config` ou pela variável de ambiente.

## **HasRetryQuota**

Essa etapa controla a utilização de solicitações de repetição exigindo que um token esteja disponível no bucket da cota de repetição. Os buckets de cota de repetição são um mecanismo para evitar novas tentativas que provavelmente não serão bem-sucedidas. Essas cotas dependem do SDK, geralmente dependem do cliente e, às vezes, até dos endpoints do serviço. Os tokens de cota de repetição disponíveis são removidos quando as solicitações falham por vários motivos e reabastecidos quando são bem-sucedidas. Quando nenhum token permanece, o loop de repetição é encerrado.



## ExponentialBackoff

Para um erro que pode ser repetido, o atraso da nova tentativa é calculado usando o recuo exponencial truncado. Os SDKs usam recuo exponencial binário truncado com instabilidade. O algoritmo a seguir mostra como a quantidade de tempo de sono, em segundos, é definida para uma resposta à solicitação  $i$ :

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

No algoritmo anterior, os seguintes valores se aplicam:

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

$r = 2$

$\text{MAX\_BACKOFF} = 20$  seconds para a maioria dos SDKs. Consulte o guia específico do SDK ou o código-fonte para confirmação.

## Compatibilidade com AWS SDKs

Os SDKs a seguir são compatíveis com os recursos e configurações descritos neste tópico.

Quaisquer exceções parciais estão anotadas. Todas as configurações de propriedade do sistema JVM são suportadas pelo AWS SDK for Java e pelo AWS SDK para Kotlin único.

SDK	C	Notas ou mais informações
<a href="#">AWS CLI v2</a>	Sim	
<a href="#">SDK para C++</a>	Sim	
<a href="#">SDK for Go V2 (1.x)</a>	Sim	
<a href="#">SDK for Go 1.x (V1)</a>	Não	
<a href="#">SDK para Java 2.x</a>	Sim	
<a href="#">SDK para Java 1.x</a>	Sim	Propriedades do sistema JVM: use <code>com.amazonaws.sdk.maxAttempts</code> em vez de <code>aws.maxAttempts</code> ; use em <code>com.amazonaws.sdk.retryMode</code> em vez de <code>aws.retryMode</code>

SDK	Compartilhado	Notas ou mais informações
<a href="#">SDK para 3.x JavaScript</a>	Sim	
<a href="#">SDK para 2.x JavaScript</a>	Não	Suporta um número máximo de novas tentativas, recuo exponencial com instabilidade e a opção de um método personalizado para recuar novamente.
<a href="#">SDK for Kotlin</a>	Sim	
<a href="#">SDK for .NET 3.x</a>	Sim	
<a href="#">SDK for PHP 3.x</a>	Sim	
<a href="#">SDK for Python (Boto3)</a>	Sim	
<a href="#">SDK for Ruby 3.x</a>	Sim	
<a href="#">SDK for Rust</a>	Sim	
<a href="#">Ferramentas para PowerShell</a>	Sim	

## Compactação de solicitações

AWS SDKs e ferramentas podem compactar cargas automaticamente ao enviar solicitações para Serviços da AWS que suportem o recebimento de cargas comprimidas. Compactar a carga útil do cliente antes de enviá-la para um serviço pode reduzir o número geral de solicitações e a largura de banda necessárias para enviar dados ao serviço, bem como reduzir as solicitações malsucedidas devido às limitações do serviço no tamanho da carga útil. Para compactação, o SDK ou a ferramenta seleciona um algoritmo de codificação compatível com o serviço e o SDK. No entanto, a lista atual de codificações possíveis consiste apenas em gzip, mas pode se expandir no futuro.

A compactação de solicitações pode ser especialmente útil se seu aplicativo estiver usando a [Amazon CloudWatch](#). CloudWatch é um serviço de monitoramento e observabilidade que coleta dados operacionais e de monitoramento na forma de registros, métricas e eventos. Um exemplo de operação de serviço que oferece suporte à compactação CloudWatch é o método de [PutMetricDataAPI](#).

Configure essa funcionalidade usando o seguinte:

**disable\_request\_compression**- configuração de AWS **config** arquivo compartilhado, **AWS\_DISABLE\_REQUEST\_COMPRESSION**: variável de ambiente, **aws.disableRequestCompression**- Propriedade do sistema JVM: somente Java/Kotlin

Ativa ou desativa se o SDK ou a ferramenta compactarão uma carga antes de enviar uma solicitação.

Valor padrão: `false`

Valores válidos:

- **true** – Desative a compactação de solicitações.
- **false** – Use a compactação de solicitações quando possível.

**request\_min\_compression\_size\_bytes**- configuração de AWS **config** arquivo compartilhado, **AWS\_REQUEST\_MIN\_COMPRESSION\_SIZE\_BYTES**: variável de ambiente, **aws.requestMinCompressionSizeBytes**- Propriedade do sistema JVM: somente Java/Kotlin

Define o tamanho mínimo em bytes do corpo da solicitação que o SDK ou a ferramenta devem compactar. Cargas pequenas podem ficar maiores quando compactadas, portanto, há um limite mínimo em que faz sentido realizar a compactação. Esse valor é inclusivo, um tamanho de solicitação maior que ou igual ao valor é compactado.

Valor padrão: 10240 bytes

Valores válidos: valor inteiro entre 0 e 10485760 bytes, inclusive.

## Compatibilidade com AWS SDKs

Os SDKs a seguir são compatíveis com os recursos e configurações descritos neste tópico. Quaisquer exceções parciais estão anotadas. Todas as configurações de propriedade do sistema JVM são suportadas pelo AWS SDK for Java e pelo AWS SDK para Kotlin único.

SDK	Compatibilidade	Notas ou mais informações
<a href="#">AWS CLI v2</a>	Sim	
<a href="#">SDK para C++</a>	Sim	

SDK	Comentários ou mais informações
<a href="#">SDK para Go V2 (1.x)</a>	Sim
<a href="#">SDK para Go 1.x (V1)</a>	Não
<a href="#">SDK para Java 2.x</a>	Sim
<a href="#">SDK para Java 1.x</a>	Não
<a href="#">SDK para 3.x JavaScript</a>	Sim
<a href="#">SDK para 2.x JavaScript</a>	Não
<a href="#">SDK para Kotlin</a>	Sim
<a href="#">SDK para .NET 3.x</a>	Sim
<a href="#">SDK para PHP 3.x</a>	Sim
<a href="#">SDK para Python (Boto3)</a>	Sim
<a href="#">SDK para Ruby 3.x</a>	Sim
<a href="#">SDK para Rust</a>	Sim
<a href="#">Ferramentas para PowerShell</a>	Sim

## Endpoints específicos de serviço

A configuração de endpoint específico de serviço oferece a opção de usar um endpoint de sua escolha para solicitações de API e para ter a persistência dessa escolha. Essas configurações oferecem flexibilidade para permitir endpoints locais, endpoints da VPC e ambientes de desenvolvimento da AWS local de terceiros. Diferentes endpoints podem ser usados para ambientes de teste e produção. Você pode especificar um URL de endpoint para Serviços da AWS individuais.

Configure essa funcionalidade usando o seguinte:

**endpoint\_url**- configuração de AWS **config** arquivo compartilhado, **AWS\_ENDPOINT\_URL**: variável de ambiente, **aws.endpointUrl**- Propriedade do sistema JVM: somente Java/Kotlin

Quando especificada diretamente em um perfil ou como uma variável de ambiente, esta configuração especifica o endpoint usado para todas as solicitações de serviço. Este endpoint é substituído por qualquer endpoint específico do serviço configurado.

Você também pode usar essa configuração em uma `services` seção de um AWS `config` arquivo compartilhado para definir um endpoint personalizado para um serviço específico. Para obter uma lista de todas as chaves de identificação de serviço a serem usadas nas subseções dentro da seção `services`, consulte [Identificadores para endpoints específicos de serviço](#).

Valor padrão: none

Valores válidos: um URL incluindo o esquema e o host do endpoint. Opcionalmente, o URL pode conter um componente de caminho que contenha um ou mais segmentos de caminho.

**AWS\_ENDPOINT\_URL\_<SERVICE>**: variável de ambiente, **aws.endpointUrl<ServiceName>**- Propriedade do sistema JVM: somente Java/Kotlin

**AWS\_ENDPOINT\_URL\_<SERVICE>**, onde `<SERVICE>` está o AWS service (Serviço da AWS) identificador, define um endpoint personalizado para um serviço específico. Para obter uma lista de todas as variáveis de ambiente específicas do serviço, consulte [Identificadores para endpoints específicos de serviço](#).

Este endpoint específico do serviço substitui qualquer endpoint global configurado em **AWS\_ENDPOINT\_URL**.

Valor padrão: none

Valores válidos: um URL incluindo o esquema e o host do endpoint. Opcionalmente, o URL pode conter um componente de caminho que contenha um ou mais segmentos de caminho.

**ignore\_configured\_endpoint\_urls**- configuração de AWS **config** arquivo compartilhado, **AWS\_IGNORE\_CONFIGURED\_ENDPOINT\_URLS**: variável de ambiente, **aws.ignoreConfiguredEndpointUrls**- Propriedade do sistema JVM: somente Java/Kotlin

Esta configuração é usada para ignorar todas as configurações personalizadas de endpoints.

Observe que qualquer endpoint explícito definido no código ou no próprio cliente de serviço é usado independentemente desta configuração. Por exemplo, incluir o parâmetro da linha de `--endpoint-url` comando com um AWS CLI comando ou passar uma URL de endpoint para um construtor de cliente sempre terá efeito.

Valor padrão: `false`

Valores válidos:

- **true**: o SDK ou a ferramenta não lê nenhuma opção de configuração personalizada do arquivo `config` compartilhado ou das variáveis de ambiente para definir um URL de endpoint.
- **false**: o SDK ou a ferramenta usa todos os endpoints disponíveis fornecidos pelo usuário a partir do arquivo `config` compartilhado ou de variáveis de ambiente.

## Configurar endpoints usando variáveis de ambiente

Para rotear solicitações de todos os serviços para um URL de endpoint personalizado, defina a variável de ambiente global `AWS_ENDPOINT_URL`.

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

Para encaminhar solicitações de um URL específico AWS service (Serviço da AWS) para um endpoint personalizado, use a variável de `AWS_ENDPOINT_URL_<SERVICE>` ambiente. Amazon DynamoDB tem um `serviceId` de [DynamoDB](#). Para esse serviço, a variável de ambiente do URL do endpoint é `AWS_ENDPOINT_URL_DYNAMODB`. Este endpoint tem precedência sobre o endpoint global definido em `AWS_ENDPOINT_URL` para este serviço.

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

Como outro exemplo, AWS Elastic Beanstalk tem um `serviceId` de [Elastic Beanstalk](#). O AWS service (Serviço da AWS) identificador é baseado no modelo de API, substituindo todos os espaços `serviceId` por sublinhados e colocando todas as letras em maiúsculas. Para configurar o endpoint para este serviço, a variável de ambiente correspondente é `AWS_ENDPOINT_URL_ELASTIC_BEANSTALK`. Para obter uma lista de todas as variáveis de ambiente específicas do serviço, consulte [Identificadores para endpoints específicos de serviço](#).

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

## Configurar endpoints usando o arquivo compartilhado **config**

No arquivo compartilhado `config`, `endpoint_url` é usado em locais diferentes para diferentes funcionalidades.

- `endpoint_url` especificado diretamente em um `profile` torna esse endpoint no endpoint global.
- `endpoint_url` aninhado sob uma chave identificadora de serviço em uma seção `services`, faz com que esse endpoint se aplique às solicitações feitas somente para esse serviço. Para obter detalhes sobre como definir uma seção `services` no arquivo compartilhado [Formato do arquivo de configuração](#), consulte `config`.

O exemplo a seguir usa uma definição `services` para configurar um URL de endpoint específico do serviço para o Amazon S3 e um endpoint global personalizado para ser usado para todos os demais serviços:

```
[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
```

Um único perfil pode configurar endpoints para vários serviços. Este exemplo mostra como definir os URLs de endpoint específicos do serviço para o Amazon S3 e no mesmo perfil. AWS Elastic Beanstalk tem um `serviceId` de [Elastic Beanstalk](#). O AWS service (Serviço da AWS) identificador é baseado no modelo de API, substituindo todos os espaços `serviceId` por sublinhados e colocando todas as letras em minúsculas. Assim, a chave identificadora de serviço se torna `elastic_beanstalk` e as configurações deste serviço começam na linha `elastic_beanstalk =`. Para obter uma lista de todas as chaves de identificação de serviço a serem usadas na seção `services`, consulte [Identificadores para endpoints específicos de serviço](#).

```
[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
```

A seção de configuração de serviço pode ser usada a partir de vários perfis. Por exemplo, dois perfis podem usar a mesma definição `services` ao alterar outras propriedades do perfil:

```
[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
```

## Configure endpoints em perfis usando credenciais baseadas em funções

Se o seu perfil tiver credenciais baseadas em perfis configurados por meio de um parâmetro `source_profile` para a funcionalidade “assumir função” do IAM, o SDK usará somente configurações de serviço para o perfil especificado. Ele não usa perfis com funções vinculadas a ele. Por exemplo, usando o seguinte arquivo compartilhado config:

```
[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 você usar o perfil B e fizer uma chamada em seu código para o Amazon EC2, o endpoint será resolvido como `https://profile-b-ec2-endpoint.aws`. Se o seu código fizer uma solicitação para qualquer outro serviço, a resolução do endpoint não seguirá nenhuma lógica personalizada. O endpoint não é resolvido para o endpoint global definido no perfil A. Para que um endpoint global tenha efeito para o perfil B, você precisaria configurar `endpoint_url` diretamente no perfil B.



Para obter mais informações sobre a configuração `source_profile`, consulte [Assuma o perfil de provedor de credenciais](#).

## Precedência de configurações

As configurações deste atributo podem ser usadas ao mesmo tempo, mas somente um valor terá prioridade por serviço. Para chamadas de API feitas para um determinado AWS service (Serviço da AWS), a seguinte ordem é usada para selecionar um valor:

1. Qualquer configuração explícita definida no código ou no próprio cliente de serviço tem precedência sobre qualquer outra coisa.
  - Para o AWS CLI, esse é o valor fornecido pelo parâmetro da linha de `--endpoint-url` comando. Para um SDK, as atribuições explícitas podem assumir a forma de um parâmetro que você define ao instanciar um AWS service (Serviço da AWS) cliente ou objeto de configuração.
2. O valor fornecido por uma variável de ambiente específica do serviço, como `AWS_ENDPOINT_URL_DYNAMODB`.
3. O valor fornecido pela variável de ambiente global do endpoint `AWS_ENDPOINT_URL`.
4. O valor fornecido pela configuração `endpoint_url` aninhada em uma chave identificadora de serviço em uma seção `services` do arquivo compartilhado `config`.
5. O valor fornecido pela configuração `endpoint_url` especificado diretamente em um `profile` do arquivo compartilhado `config`.
6. Qualquer URL de endpoint padrão para o respectivo AWS service (Serviço da AWS) é usada por último.

## Compatibilidade com AWS SDKs

Os SDKs a seguir são compatíveis com os recursos e configurações descritos neste tópico. Quaisquer exceções parciais estão anotadas. Todas as configurações de propriedade do sistema JVM são suportadas pelo AWS SDK for Java e pelo AWS SDK para Kotlin único.

SDK	Compatível	Notas ou mais informações
<a href="#">AWS CLI v2</a>	Sim	
<a href="#">SDK para C++</a>	Não	

SDK	Compartilhado	Notas ou mais informações
<a href="#">SDK para Go V2 (1.x)</a>	Sim	
<a href="#">SDK para Go 1.x (V1)</a>	Não	
<a href="#">SDK para Java 2.x</a>	Não	
<a href="#">SDK para Java 1.x</a>	Não	
<a href="#">SDK para 3.x JavaScript</a>	Sim	
<a href="#">SDK para 2.x JavaScript</a>	Não	
<a href="#">SDK para Kotlin</a>	Sim	
<a href="#">SDK para .NET 3.x</a>	Sim	
<a href="#">SDK para PHP 3.x</a>	Sim	
<a href="#">SDK para Python (Boto3)</a>	Sim	
<a href="#">SDK para Ruby 3.x</a>	Sim	
<a href="#">SDK para Rust</a>	Não	
<a href="#">Ferramentas para PowerShell</a>	Sim	

## Identificadores para endpoints específicos de serviço

Para obter informações sobre como e onde usar os identificadores na tabela a seguir, consulte [Endpoints específicos de serviço](#).

<b>serviceId</b>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</b>
AccessAnalyzer	ac	AWS_ENDPOINT_URL_ACCESSANALYZER	
Account	ac	AWS_ENDPOINT_URL_ACCOUNT	
ACM	ac	AWS_ENDPOINT_URL_ACM	
ACM PCA	ac	AWS_ENDPOINT_URL_ACM_PCA	
Alexa For Business	af	AWS_ENDPOINT_URL_ALEXA_FOR_BUSINESS	
amp	ar	AWS_ENDPOINT_URL_AMP	
Amplify	ar	AWS_ENDPOINT_URL_AMPLIFY	
AmplifyBackend	ar	AWS_ENDPOINT_URL_AMPLIFYBACKEND	
AmplifyUIBuilder	ar	AWS_ENDPOINT_URL_AMPLIFYUIBUILDER	
API Gateway	ap	AWS_ENDPOINT_URL_API_GATEWAY	

<b>serviceId</b>	Cl	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	variável de ambiente
ApiGatewayManagem entApi	ap y n	AWS_ENDPOINT_URL_APIGATEWAYMANAGEMENTAPI	
ApiGatewayV2	ap y	AWS_ENDPOINT_URL_APIGATEWAYV2	
AppConfig	ap	AWS_ENDPOINT_URL_APPCONFIG	
AppConfigData	ap d	AWS_ENDPOINT_URL_APPCONFIGDATA	
AppFabric	ap	AWS_ENDPOINT_URL_APPFABRIC	
Appflow	ap	AWS_ENDPOINT_URL_APPFLOW	
AppIntegrations	ap a	AWS_ENDPOINT_URL_APPINTEGRATIONS	
Application Auto Scaling	ap o c	AWS_ENDPOINT_URL_APPLICATION_AUTO_SCALING	

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

<b>serviceId</b>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</b>
	id ac de se pa Al co ar cc ha		
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	te
BackupStorage	b:	AWS_ENDPOINT_URL_BACKUPSTORAGE	ri
Batch	b:	AWS_ENDPOINT_URL_BATCH	
BCM Data Exports	b:	AWS_ENDPOINT_URL_BCM_DATA_EXPORTS	e:
Bedrock	b:	AWS_ENDPOINT_URL_BEDROCK	
Bedrock Agent	b:	AWS_ENDPOINT_URL_BEDROCK_AGENT	ge

<b>serviceId</b>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</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>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	variável de ambiente
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>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	variável de ambiente
CloudDirectory	c: AWS_ENDPOINT_URL_CLOUDDIRECTORY	
CloudFormation	c: AWS_ENDPOINT_URL_CLOUDFORMATION	
CloudFront	c: AWS_ENDPOINT_URL_CLOUDFRONT	
CloudFront KeyValuesStore	c: AWS_ENDPOINT_URL_CLOUDFRONT_KEYVALUESTORE	
CloudHSM	c: AWS_ENDPOINT_URL_CLOUDHSM	
CloudHSM V2	c: AWS_ENDPOINT_URL_CLOUDHSM_V2	
CloudSearch	c: AWS_ENDPOINT_URL_CLOUDSEARCH	
CloudSearch Domain	c: AWS_ENDPOINT_URL_CLOUDSEARCH_DOMAIN	

<b>serviceId</b>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</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>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</b>
CodeGuru Security	cc	AWS_ENDPOINT_URL_CODEGURU_SECURITY	
CodeGuruProfiler	cc	AWS_ENDPOINT_URL_CODEGURUPROFILER	
CodePipeline	cc	AWS_ENDPOINT_URL_CODEPIPELINE	
CodeStar	cc	AWS_ENDPOINT_URL_CODESTAR	
CodeStar connections	cc	AWS_ENDPOINT_URL_CODESTAR_CONNECTIONS	
codestar notificat ions	cc	AWS_ENDPOINT_URL_CODESTAR_NOTIFICATIONS	
Cognito Identity	cc	AWS_ENDPOINT_URL_COGNITO_IDENTITY	
Cognito Identity Provider	cc	AWS_ENDPOINT_URL_COGNITO_IDENTITY_PROVIDER	

<b>serviceId</b>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</b>
Cognito Sync	co	AWS_ENDPOINT_URL_COGNITO_SYNC	
Comprehend	co	AWS_ENDPOINT_URL_COMPREHEND	
ComprehendMedical	co	AWS_ENDPOINT_URL_COMPREHENDMEDICAL	
Compute Optimizer	co	AWS_ENDPOINT_URL_COMPUTE_OPTIMIZER	
Config Service	co	AWS_ENDPOINT_URL_CONFIG_SERVICE	
Connect	co	AWS_ENDPOINT_URL_CONNECT	
Connect Contact Lens	co	AWS_ENDPOINT_URL_CONNECT_CONTACT_LENS	
ConnectCampaigns	co	AWS_ENDPOINT_URL_CONNECTCAMPAIGNS	
ConnectCases	co	AWS_ENDPOINT_URL_CONNECTCASES	

<b>serviceId</b>	<pre> AWS_ENDPOINT_URL_&lt;SERVICE&gt; </pre>	variável de ambiente
ConnectParticipant	<pre> AWS_ENDPOINT_URL_CONNECTPARTICIPANT </pre>	
ControlTower	<pre> AWS_ENDPOINT_URL_CONTROLTOWER </pre>	
Cost Optimization Hub	<pre> AWS_ENDPOINT_URL_COST_OPTIMIZATION_HUB </pre>	
Cost and Usage Report Service	<pre> AWS_ENDPOINT_URL_COST_AND_USAGE_REPO RT_SERVICE </pre>	
Customer Profiles	<pre> AWS_ENDPOINT_URL_CUSTOMER_PROFILES </pre>	
DataBrew	<pre> AWS_ENDPOINT_URL_DATABREW </pre>	
DataExchange	<pre> AWS_ENDPOINT_URL_DATAEXCHANGE </pre>	
Data Pipeline	<pre> AWS_ENDPOINT_URL_DATA_PIPELINE </pre>	

<b>serviceId</b>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</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:	AWS_ENDPOINT_URL_DEVICE_FARM	
DevOps Guru	d:	AWS_ENDPOINT_URL_DEVOPS_GURU	
Direct Connect	d:	AWS_ENDPOINT_URL_DIRECT_CONNECT	
Application Discovery Service	a:	AWS_ENDPOINT_URL_APPLICATION_DISCOVERY_SERVICE	
DLM	d:	AWS_ENDPOINT_URL_DLM	

<b>serviceId</b>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</b>
Database Migration Service	d:	AWS_ENDPOINT_URL_DATABASE_MIGRATION_	
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>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</b>
ECR	e	AWS_ENDPOINT_URL_ECR	
ECR PUBLIC	e	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	AWS_ENDPOINT_URL_ELASTIC_INFERENCE	
ElastiCache	e	AWS_ENDPOINT_URL_ELASTICACHE	
Elastic Beanstalk	e	AWS_ENDPOINT_URL_ELASTIC_BEANSTALK	
Elastic Transcoder	e	AWS_ENDPOINT_URL_ELASTIC_TRANSCODER	



<b>serviceId</b>	Cl	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	variável de ambiente
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>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</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	fc	AWS_ENDPOINT_URL_FORECAST	
forecastquery	fc	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>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</b>
Glacier	g:	AWS_ENDPOINT_URL_GLACIER	
Global Accelerator	g:	AWS_ENDPOINT_URL_GLOBAL_ACCELERATOR	
Glue	g:	AWS_ENDPOINT_URL_GLUE	
grafana	g:	AWS_ENDPOINT_URL_GRAFANA	
Greengrass	g:	AWS_ENDPOINT_URL_GREENGRASS	
GreengrassV2	g:	AWS_ENDPOINT_URL_GREENGRASSV2	
GroundStation	g:	AWS_ENDPOINT_URL_GROUNDSTATION	
GuardDuty	g:	AWS_ENDPOINT_URL_GUARDDUTY	
Health	h:	AWS_ENDPOINT_URL_HEALTH	
HealthLake	h:	AWS_ENDPOINT_URL_HEALTHLAKE	

<b>serviceId</b>	Cl id ac de se pa Al co ar cc ha	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b> variável de ambiente
Honeycode	hc	AWS_ENDPOINT_URL_HONEYCODE
IAM	ia	AWS_ENDPOINT_URL_IAM
identitystore	id to	AWS_ENDPOINT_URL_IDENTITYSTORE
imagebuilder	ib de	AWS_ENDPOINT_URL_IMAGEBUILDER
ImportExport	ie o:	AWS_ENDPOINT_URL_IMPORTEXPORT
Inspector	in	AWS_ENDPOINT_URL_INSPECTOR
Inspector Scan	in _s	AWS_ENDPOINT_URL_INSPECTOR_SCAN
Inspector2	in 2	AWS_ENDPOINT_URL_INSPECTOR2
InternetMonitor	im oi	AWS_ENDPOINT_URL_INTERNETMONITOR
IoT	io	AWS_ENDPOINT_URL_IOT

<b>serviceId</b>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</b>
IoT Data Plane	id	AWS_ENDPOINT_URL_IOT_DATA_PLANE	
IoT Jobs Data Plane	id	AWS_ENDPOINT_URL_IOT_JOBS_DATA_PLANE	
IoT 1Click Devices Service	id	AWS_ENDPOINT_URL_IOT_1CLICK_DEVICES_SERVICE	
IoT 1Click Projects	id	AWS_ENDPOINT_URL_IOT_1CLICK_PROJECTS	
IoTAnalytics	id	AWS_ENDPOINT_URL_IOTANALYTICS	
IotDeviceAdvisor	id	AWS_ENDPOINT_URL_IOTDEVICEADVISOR	
IoT Events	id	AWS_ENDPOINT_URL_IOT_EVENTS	
IoT Events Data	id	AWS_ENDPOINT_URL_IOT_EVENTS_DATA	

<b>serviceId</b>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</b>
IoT FleetHub	id	AWS_ENDPOINT_URL_IOTFLEETHUB	
IoT FleetWise	id	AWS_ENDPOINT_URL_IOTFLEETWISE	
IoT Secure Tunneling	id	AWS_ENDPOINT_URL_IOTSECURETUNNELING	
IoT SiteWise	id	AWS_ENDPOINT_URL_IOTSITWISE	
IoT ThingsGraph	id	AWS_ENDPOINT_URL_IOTTHINGSGRAPH	
IoT TwinMaker	id	AWS_ENDPOINT_URL_IOTTWINMAKER	
IoT Wireless	id	AWS_ENDPOINT_URL_IOT_WIRELESS	
ivs	id	AWS_ENDPOINT_URL_IVS	
IVS RealTime	id	AWS_ENDPOINT_URL_IVS_REALTIME	

<b>serviceId</b>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</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>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</b>
Kinesis Video Signaling	id ac de se pa Al co ar cc ha	k: AWS_ENDPOINT_URL_KINESIS_VIDEO_SIGNALING	
Kinesis Video WebRTC Storage	id t e	k: AWS_ENDPOINT_URL_KINESIS_VIDEO_WEBRTC_STORAGE	
Kinesis Analytics	na	k: AWS_ENDPOINT_URL_KINESIS_ANALYTICS	
Kinesis Analytics V2	na v:	k: AWS_ENDPOINT_URL_KINESIS_ANALYTICS_V2	
Kinesis Video	id	k: AWS_ENDPOINT_URL_KINESIS_VIDEO	
KMS	kr	AWS_ENDPOINT_URL_KMS	
LakeFormation	t:	l: AWS_ENDPOINT_URL_LAKEFORMATION	
Lambda	l:	AWS_ENDPOINT_URL_LAMBDA	



<b>serviceId</b>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</b>
Launch Wizard	l	AWS_ENDPOINT_URL_LAUNCH_WIZARD	
Lex Model Building Service	l	AWS_ENDPOINT_URL_LEX_MODEL_BUILDING_	
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_LIN	

<b>serviceId</b>	Cl id ac de se pa Al co ar co ha	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b> variável de ambiente
License Manager User Subscriptions	l a e: i)	1: AWS_ENDPOINT_URL_LICENSE_MANAGER_USER_SUBSCRIPTIONS
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>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</b>
Machine Learning	m:	AWS_ENDPOINT_URL_MACHINE_LEARNING	
Macie2	m:	AWS_ENDPOINT_URL_MACIE2	
ManagedBlockchain	m:	AWS_ENDPOINT_URL_MANAGEDBLOCKCHAIN	
ManagedBlockchain Query	m:	AWS_ENDPOINT_URL_MANAGEDBLOCKCHAIN_QUERY	
Marketplace Agreement	m:	AWS_ENDPOINT_URL_MARKETPLACE_AGREEMENT	
Marketplace Catalog	m:	AWS_ENDPOINT_URL_MARKETPLACE_CATALOG	
Marketplace Deployment	m:	AWS_ENDPOINT_URL_MARKETPLACE_DEPLOYMENT	

<b>serviceId</b>	<p>Cl</p> <p>id</p> <p>ac</p> <p>de</p> <p>se</p> <p>pa</p> <p>Al</p> <p>co</p> <p>ar</p> <p>cc</p> <p>ha</p>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b> variável de ambiente
Marketplace Entitlement Service	<p>m</p> <p>c</p> <p>e</p> <p>v:</p>	<b>AWS_ENDPOINT_URL_MARKETPLACE_ENTITLEMENT_SERVICE</b>
Marketplace Commerce Analytics	<p>m</p> <p>c</p> <p>c</p> <p>i</p>	<b>AWS_ENDPOINT_URL_MARKETPLACE_COMMERCIAL_ANALYTICS</b>
MediaConnect	<p>m</p> <p>e</p>	<b>AWS_ENDPOINT_URL_MEDIACONNECT</b>
MediaConvert	<p>m</p> <p>e:</p>	<b>AWS_ENDPOINT_URL_MEDIACONVERT</b>
MediaLive	<p>m</p>	<b>AWS_ENDPOINT_URL_MEDIALIVE</b>
MediaPackage	<p>m</p> <p>a</p>	<b>AWS_ENDPOINT_URL_MEDIAPACKAGE</b>
MediaPackage Vod	<p>m</p> <p>a</p>	<b>AWS_ENDPOINT_URL_MEDIAPACKAGE_VOD</b>

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

<b>serviceId</b>	Cl id ac de se pa Al co ar cc ha	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b> variável de ambiente
Migration Hub Refactor Spaces	m: c: e:	AWS_ENDPOINT_URL_MIGRATION_HUB_REFAC TOR_SPACES
MigrationHub Config	h: g:	AWS_ENDPOINT_URL_MIGRATIONHUB_CONFIG
MigrationHubOrches trator	h: t:	AWS_ENDPOINT_URL_MIGRATIONHUBORCHESTRATOR
MigrationHubStrategy	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>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</b>
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>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</b>
	id ac de se pa Al co ar cc ha		
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	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	variável de ambiente
Pca Connector Ad	pc	AWS_ENDPOINT_URL_PCA_CONNECTOR_AD	
Personalize	pe	AWS_ENDPOINT_URL_PERSONALIZE	
Personalize Events	pe	AWS_ENDPOINT_URL_PERSONALIZE_EVENTS	
Personalize Runtime	pe	AWS_ENDPOINT_URL_PERSONALIZE_RUNTIME	
PI	p:	AWS_ENDPOINT_URL_PI	
Pinpoint	p:	AWS_ENDPOINT_URL_PINPOINT	
Pinpoint Email	p:	AWS_ENDPOINT_URL_PINPOINT_EMAIL	
Pinpoint SMS Voice	p:	AWS_ENDPOINT_URL_PINPOINT_SMS_VOICE	

<b>serviceId</b>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</b>
Pinpoint SMS Voice V2	p:	AWS_ENDPOINT_URL_PINPOINT_SMS_VOICE_V2	
Pipes	p:	AWS_ENDPOINT_URL_PIPES	
Polly	p:	AWS_ENDPOINT_URL_POLLY	
Pricing	p:	AWS_ENDPOINT_URL_PRICING	
PrivateNetworks	p:	AWS_ENDPOINT_URL_PRIVATENETWORKS	
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	
QuickSight	q:	AWS_ENDPOINT_URL_QUICKSIGHT	

<b>serviceId</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</b>
RAM	<a href="#">AWS_ENDPOINT_URL_RAM</a>	
rbn	<a href="#">AWS_ENDPOINT_URL_RBIN</a>	
RDS	<a href="#">AWS_ENDPOINT_URL_RDS</a>	
RDS Data	<a href="#">AWS_ENDPOINT_URL_RDS_DATA</a>	
Redshift	<a href="#">AWS_ENDPOINT_URL_REDSHIFT</a>	
Redshift Data	<a href="#">AWS_ENDPOINT_URL_REDSHIFT_DATA</a>	
Redshift Serverless	<a href="#">AWS_ENDPOINT_URL_REDSHIFT_SERVERLESS</a>	
Rekognition	<a href="#">AWS_ENDPOINT_URL_REKOGNITION</a>	
repostspace	<a href="#">AWS_ENDPOINT_URL_REPOSTSPACE</a>	
resiliencehub	<a href="#">AWS_ENDPOINT_URL_RESILIENCEHUB</a>	

<b>serviceId</b>	Cl id ac de se pa Al co ar cc ha	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b> variável de ambiente
Resource Explorer 2	r e 2	<b>AWS_ENDPOINT_URL_RESOURCE_EXPLORER_2</b>
Resource Groups	r g	<b>AWS_ENDPOINT_URL_RESOURCE_GROUPS</b>
Resource Groups Tagging API	r g g	<b>AWS_ENDPOINT_URL_RESOURCE_GROUPS_TAGGING_API</b>
RoboMaker	r	<b>AWS_ENDPOINT_URL_ROBOMAKER</b>
RolesAnywhere	r h	<b>AWS_ENDPOINT_URL_ROLESEANYWHERE</b>
Route 53	r	<b>AWS_ENDPOINT_URL_ROUTE_53</b>
Route53 Recovery Cluster	r e l	<b>AWS_ENDPOINT_URL_ROUTE53_RECOVERY_CLUSTER</b>

<b>serviceId</b>	Cl	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	variável de ambiente
Route53 Recovery Control Config	id ac de se pa Al co ar cc ha	AWS_ENDPOINT_URL_ROUTE53_RECOVERY_CO NTROL_CONFIG	
Route53 Recovery Readiness	id ac de se pa Al co ar cc ha	AWS_ENDPOINT_URL_ROUTE53_RECOVERY_RE ADINESS	
Route 53 Domains	id ac de se pa Al co ar cc ha	AWS_ENDPOINT_URL_ROUTE_53_DOMAINS	
Route53Resolver	id ac de se pa Al co ar cc ha	AWS_ENDPOINT_URL_ROUTE53RESOLVER	
RUM	id ac de se pa Al co ar cc ha	AWS_ENDPOINT_URL_RUM	
S3	id ac de se pa Al co ar cc ha	AWS_ENDPOINT_URL_S3	
S3 Control	id ac de se pa Al co ar cc ha	AWS_ENDPOINT_URL_S3_CONTROL	
S3Outposts	id ac de se pa Al co ar cc ha	AWS_ENDPOINT_URL_S3OUTPOSTS	

<b>serviceId</b>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</b>
SageMaker	id ac de se pa Al co ar cc ha	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 id ac de se pa Al co ar cc ha	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b> variável de ambiente
savingsplans	s a	AWS_ENDPOINT_URL_SAVINGSPLANS
Scheduler	s	AWS_ENDPOINT_URL_SCHEDULER
schemas	s	AWS_ENDPOINT_URL_SCHEMAS
SimpleDB	s	AWS_ENDPOINT_URL_SIMPLEDB
Secrets Manager	s a	AWS_ENDPOINT_URL_SECRETS_MANAGER
SecurityHub	s u	AWS_ENDPOINT_URL_SECURITYHUB
SecurityLake	s a	AWS_ENDPOINT_URL_SECURITYLAKE
ServerlessApplicat ionRepository	s s i t	AWS_ENDPOINT_URL_SERVERLESSAPPLICATI ONREPOSITORY
Service Quotas	s u	AWS_ENDPOINT_URL_SERVICE_QUOTAS

<b>serviceId</b>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</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	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	variável de ambiente
Snowball	s:	AWS_ENDPOINT_URL_SNOWBALL	
SNS	s:	AWS_ENDPOINT_URL_SNS	
SQS	s:	AWS_ENDPOINT_URL_SQS	
SSM	s:	AWS_ENDPOINT_URL_SSM	
SSM Contacts	s:	AWS_ENDPOINT_URL_SSM_CONTACTS	
SSM Incidents	s:	AWS_ENDPOINT_URL_SSM_INCIDENTS	
Ssm Sap	s:	AWS_ENDPOINT_URL_SSM_SAP	
SSO	s:	AWS_ENDPOINT_URL_SSO	
SSO Admin	s:	AWS_ENDPOINT_URL_SSO_ADMIN	
SSO OIDC	s:	AWS_ENDPOINT_URL_SSO_OIDC	
SFN	s:	AWS_ENDPOINT_URL_SFN	
Storage Gateway	s:	AWS_ENDPOINT_URL_STORAGE_GATEWAY	

<b>serviceId</b>	Classe	Nome da variável de ambiente
	Client	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b> variável de ambiente
STS	STSClient	AWS_ENDPOINT_URL_STS
SupplyChain	SupplyChainClient	AWS_ENDPOINT_URL_SUPPLYCHAIN
Support	SupportClient	AWS_ENDPOINT_URL_SUPPORT
Support App	SupportAppClient	AWS_ENDPOINT_URL_SUPPORT_APP
SWF	SWFClient	AWS_ENDPOINT_URL_SWF
synthetics	SyntheticsClient	AWS_ENDPOINT_URL_SYNTHETICS
Textract	TextractClient	AWS_ENDPOINT_URL_TEXTRACT
Timestream InfluxDB	TimestreamInfluxDBClient	AWS_ENDPOINT_URL_TIMESTREAM_INFLUXDB
Timestream Query	TimestreamQueryClient	AWS_ENDPOINT_URL_TIMESTREAM_QUERY
Timestream Write	TimestreamWriteClient	AWS_ENDPOINT_URL_TIMESTREAM_WRITE

<b>serviceId</b>	Cl	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	variável de ambiente
tnb	ti	AWS_ENDPOINT_URL_TNB	
Transcribe	t:	AWS_ENDPOINT_URL_TRANSCRIBE	
Transfer	t:	AWS_ENDPOINT_URL_TRANSFER	
Translate	t:	AWS_ENDPOINT_URL_TRANSLATE	
TrustedAdvisor	t:	AWS_ENDPOINT_URL_TRUSTEDADVISOR	
VerifiedPermissions	v:	AWS_ENDPOINT_URL_VERIFIEDPERMISSIONS	
Voice ID	v:	AWS_ENDPOINT_URL_VOICE_ID	
VPC Lattice	v:	AWS_ENDPOINT_URL_VPC_LATTICE	
WAF	w:	AWS_ENDPOINT_URL_WAF	
WAF Regional	w:	AWS_ENDPOINT_URL_WAF_REGIONAL	

<b>serviceId</b>	<b>Cl</b>	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	<b>variável de ambiente</b>
WAFV2	wi	AWS_ENDPOINT_URL_WAFV2	
WellArchitected	wi	AWS_ENDPOINT_URL_WELLARCHITECTED	
Wisdom	wi	AWS_ENDPOINT_URL_WISDOM	
WorkDocs	wi	AWS_ENDPOINT_URL_WORKDOCS	
WorkLink	wi	AWS_ENDPOINT_URL_WORKLINK	
WorkMail	wi	AWS_ENDPOINT_URL_WORKMAIL	
WorkMailMessageFlow	wi	AWS_ENDPOINT_URL_WORKMAILMESSAGEFLOW	
WorkSpaces	wi	AWS_ENDPOINT_URL_WORKSPACES	
WorkSpaces Thin Client	wi	AWS_ENDPOINT_URL_WORKSPACES_THIN_CLIENT	
WorkSpaces Web	wi	AWS_ENDPOINT_URL_WORKSPACES_WEB	

<b>serviceId</b>	Classe	Nome da propriedade	Tipo
	ClientConfiguration	<b>AWS_ENDPOINT_URL_&lt;SERVICE&gt;</b>	variável de ambiente
	ClientConfiguration	<b>AWS_ENDPOINT_URL_XRAY</b>	variável de ambiente

## Padrões de configuração inteligente

Com o recurso de padrões de configuração inteligente, os AWS SDKs podem fornecer valores padrão predefinidos e otimizados para outras configurações.

Configure esta funcionalidade usando o seguinte:

**defaults\_mode**- configuração de AWS **config** arquivo compartilhado, **AWS\_DEFAULTS\_MODE** - variável de ambiente, **aws.defaultsMode**- Propriedade do sistema JVM: somente Java/Kotlin

Com essa configuração, você pode escolher um modo que se alinhe à arquitetura do aplicativo e, em seguida, forneça valores padrão otimizados para o aplicativo. Se uma configuração do AWS SDK tiver um valor definido explicitamente, esse valor sempre terá precedência. Se uma configuração do AWS SDK não tiver um valor definido explicitamente e não `defaults_mode` for igual ao legado, esse recurso poderá fornecer valores padrão diferentes para várias configurações otimizadas para seu aplicativo. As configurações podem incluir o seguinte: configurações de comunicação HTTP, comportamento de repetição, configurações de endpoint regional do serviço e, potencialmente, qualquer configuração relacionada ao SDK. Os clientes que usam esse atributo podem obter novos padrões de configuração personalizados para cenários de uso comuns. Se seu `defaults_mode` não for igual a `Legacy`, recomendamos realizar testes de seu aplicativo ao atualizar o SDK, pois os valores padrão fornecidos podem mudar à medida que as melhores práticas evoluem.

Valor padrão: `Legacy`

Observação: as novas versões principais dos SDKs usarão como padrão de `standard`.

Valores válidos:

- `legacy` – Fornece configurações padrão que variam de acordo com o SDK e existiam antes do estabelecimento do `defaults_mode`.
- `standard` – Fornece os valores padrão recomendados mais recentes que devem ser executados com segurança na maioria dos cenários.
- `in-region`— Baseia-se no modo padrão e inclui otimização personalizada para aplicativos que ligam Serviços da AWS de dentro do mesmo Região da AWS.
- `cross-region`— Baseia-se no modo padrão e inclui otimização personalizada para aplicativos que fazem chamadas Serviços da AWS em uma região diferente.
- `mobile` – Baseia-se no modo padrão e inclui otimização personalizada para aplicativos móveis.
- `auto` – Baseia-se no modo padrão e inclui atributos experimentais. O SDK tenta descobrir o ambiente de runtime para determinar automaticamente as configurações apropriadas. A detecção automática é baseada em heurísticas e não fornece 100% de precisão. Se o ambiente de runtime não puder ser determinado, o modo `standard` será usado. A detecção automática pode consultar [metadados da instância e dados do usuário](#), o que pode introduzir latência. Se a latência de inicialização for fundamental para seu aplicativo, recomendamos escolher uma `defaults_mode` explícita.

Exemplo de configuração desse valor no arquivo `config`:

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

Os parâmetros a seguir podem ser otimizados com base na seleção de `defaults_mode`:

- `retryMode` – Especifica como o SDK tenta novas tentativas. Consulte [Comportamento de repetição](#).
- `stsRegionalEndpoints`— Especifica como o SDK determina o AWS service (Serviço da AWS) endpoint que ele usa para se comunicar com o AWS Security Token Service (STS). Consulte [AWS STS Endpoints regionalizados](#).
- `s3UsEast1RegionalEndpoints`— Especifica como o SDK determina o endpoint AWS de serviço que ele usa para se comunicar com o Amazon S3 da região. `us-east-1`

- `connectTimeoutInMillis` – Depois de fazer uma tentativa inicial de conexão em um soquete, a quantidade de tempo antes do tempo limite. Se o cliente não receber a conclusão do handshake de conexão, ele desiste e falhará na operação.
- `tlsNegotiationTimeoutInMillis` – O tempo máximo que um handshake TLS pode levar desde o momento em que a mensagem CLIENT HELLO é enviada até o momento em que o cliente e o servidor negociaram totalmente as cifras e trocaram as chaves.

O valor padrão para cada configuração muda dependendo da `defaults_mode` selecionada para seu aplicativo. Atualmente, esses valores são definidos da seguinte forma (sujeitos a alterações):

Parâmetro	Modo <b>standard</b>	Modo <b>in-region</b>	Modo <b>cross-region</b>	Modo <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

Por exemplo, se o `defaults_mode` que você selecionou fosse `standard`, o valor de `standard` seria atribuído a `retry_mode` (das `retry_mode` opções válidas) e o valor de `regional` seria atribuído a `stsRegionalEndpoints` (das `stsRegionalEndpoints` opções válidas).

## Compatibilidade com AWS SDKs

Os SDKs a seguir são compatíveis com os recursos e configurações descritos neste tópico. Quaisquer exceções parciais estão anotadas. Todas as configurações de propriedade do sistema JVM são suportadas pelo AWS SDK for Java e pelo AWS SDK para Kotlin único.

SDK	Compatível	Notas ou mais informações
<a href="#">AWS CLI v2</a>	Não	
<a href="#">SDK for C++</a>	Sim	Parâmetros não otimizado s:stsRegionalEndpoints , s3UsEast1RegionalEndpoints , tlsNegotiationTimeoutInMillis .
<a href="#">SDK for Go V2 (1.x)</a>	Sim	Parâmetros não otimizado s:retryMode , stsRegionalEndpoints , s3UsEast1RegionalEndpoints .
<a href="#">SDK for Go 1.x (V1)</a>	Não	
<a href="#">SDK para Java 2.x</a>	Sim	Parâmetros não otimizados: stsRegionalEndpoints .
<a href="#">SDK para Java 1.x</a>	Não	
<a href="#">SDK para 3.x JavaScript</a>	Sim	Parâmetros não otimizado s:stsRegionalEndpoints , s3UsEast1RegionalEndpoints , tlsNegotiationTimeoutInMillis . connectTimeoutInMi



SDK	Compatível	Notas ou mais informações
		llis é chamado connectionTimeout .
<a href="#">SDK para 2.x JavaScript</a>	Não	
<a href="#">SDK for Kotlin</a>	Não	
<a href="#">SDK for .NET 3.x</a>	Sim	Parâmetros não otimizado s: connectTimeoutInMi llis ,tlsNegoti ationTimeoutInMill is .
<a href="#">SDK for PHP 3.x</a>	Sim	Parâmetros não otimizado s: tlsNegotiationTime outInMillis .
<a href="#">SDK for Python (Boto3)</a>	Sim	Parâmetros não otimizado s: tlsNegotiationTime outInMillis .
<a href="#">SDK for Ruby 3.x</a>	Sim	
<a href="#">SDK for Rust</a>	Não	
<a href="#">Ferramentas para PowerShell</a>	Sim	Parâmetros não otimizado s: connectTimeoutInMi llis ,tlsNegoti ationTimeoutInMill is .

# Bibliotecas do Common Runtime (CRT) da AWS

As bibliotecas Common Runtime (CRT) da AWS são uma biblioteca base dos SDKs. O CRT é uma família modular de pacotes independentes, escrita em C. Cada pacote oferece bom desempenho e ocupa pouco espaço para as diferentes funcionalidades necessárias. Essas funcionalidades são comuns e compartilhadas em todos os SDKs, proporcionando melhor reutilização, otimização e precisão do código. Os pacotes são:

- [awslabs/aws-c-auth](#): autenticação do lado do cliente AWS (provedores de credenciais padrão e assinatura (sigv4))
- [awslabs/aws-c-cal](#): tipos criptográficos primitivos, hashes (MD5, SHA256, SHA256 HMAC), signatários, AES
- [awslabs/aws-c-common](#): estruturas de dados básicas, tipos primitivos de encadeamento/sincronização, gerenciamento de buffer, funções relacionadas ao stdlib
- [awslabs/aws-c-compression](#): algoritmos de compressão (codificação/decodificação Huffman)
- [awslabs/aws-c-event-stream](#): processamento de mensagens de fluxo de eventos (cabeçalhos, prelúdio, carga útil, crc/trailer), implementação de chamada de procedimento remoto (RPC) em fluxos de eventos
- [awslabs/aws-c-http](#): implementação de C99 das especificações do HTTP/1.1 e do HTTP/2
- [awslabs/aws-c-io](#): soquetes (TCP, UDP), DNS, canais, circuitos de eventos, canais, SSL/TLS
- [awslabs/aws-c-iot](#): implementação C99 da integração de serviços de nuvem de IoT da AWS com dispositivos
- [awslabs/aws-c-mqtt](#): protocolo de mensagens leve e padrão para a Internet das Coisas (IoT)
- [awslabs/aws-c-s3](#): implementação da biblioteca C99 para comunicação com o serviço Amazon S3, projetada para maximizar throughput em instâncias Amazon EC2 de alta largura de banda
- [awslabs/aws-c-sdkutils](#): uma biblioteca de utilitários para analisar e gerenciar perfis AWS
- [awslabs/aws-checksums](#): CRC32c e CRC32 acelerados por hardware multiplataforma com fallback a implementações de software eficientes
- [awslabs/aws-lc](#): biblioteca criptográfica de uso geral mantida pela equipe de criptografia da AWS para AWS e seus clientes, com base no código do projeto Google BoringSSL e do projeto OpenSSL
- [awslabs/s2n](#): implementação C99 dos protocolos TLS/SSL, projetados para serem pequenos e rápidos, com a segurança como prioridade

O CRT está disponível em todos os SDKs, exceto Go.

## Adicionar dependências

As bibliotecas CRT formam uma rede complexa de relacionamentos e dependências. Conhecer essas relações é útil se você precisar criar o CRT diretamente da fonte. No entanto, a maioria dos usuários acessa a funcionalidade CRT por meio do SDK de seu idioma (como AWS SDK para C++ ou AWS SDK para Java) ou do SDK de dispositivo de IoT de seu idioma (como AWS IoT SDK para C++ ou AWS IoT SDK para Java). No diagrama a seguir, a caixa Associações de CRT do Idioma se refere ao pacote que envolve as bibliotecas CRT para o SDK de um idioma específico. Essa é uma coleção de pacotes do formulário `aws-crt-*`, em que `*` é um idioma do SDK (como [aws-crt-cpp](#) ou [aws-crt-java](#)).

A seguir está uma ilustração das dependências hierárquicas das bibliotecas CRT.

# AWS Política de manutenção de SDKs e ferramentas

## Visão geral

Este documento descreve a política de manutenção de kits de desenvolvimento de AWS software (SDKs) e ferramentas, incluindo SDKs móveis e de IoT, e suas dependências subjacentes. AWS fornece regularmente aos AWS SDKs e às ferramentas atualizações que podem conter suporte para AWS APIs novas ou atualizadas, novos recursos, aprimoramentos, correções de bugs, patches de segurança ou atualizações de documentação. As atualizações também podem abordar alterações nas dependências, nos tempos de execução da linguagem e nos sistemas operacionais. As versões do SDK são publicadas em gerenciadores de pacotes (por exemplo, Maven, NuGet PyPI) e estão disponíveis como código-fonte em GitHub.

Recomendamos que os usuários continuem up-to-date com as versões do SDK para acompanhar os recursos, as atualizações de segurança e as dependências subjacentes mais recentes. O uso contínuo de uma versão não compatível do SDK não é recomendado e é feito a critério do usuário.

## Versionamento

As versões de lançamento do AWS SDK estão na forma de X.Y.Z, onde X representa a versão principal. O aumento da versão principal de um SDK indica que esse SDK passou por mudanças significativas e substanciais para oferecer suporte a novos idiomas e padrões na linguagem. As versões principais são introduzidas quando interfaces públicas (por exemplo, classes, métodos, tipos etc.), comportamentos ou semânticas mudam. Os aplicativos precisam ser atualizados para que funcionem com a versão mais recente do SDK. É importante atualizar as versões principais com cuidado e de acordo com as diretrizes de atualização fornecidas pelo AWS.

## Ciclo de vida da versão principal do SDK

O ciclo de vida das principais versões de SDKs e Ferramentas consiste em 5 fases, descritas abaixo.

- **Developer Preview (Fase 0)** - Durante essa fase, os SDKs não são suportados, não devem ser usados em ambientes de produção e são destinados apenas para fins de acesso antecipado e feedback. É possível que versões futuras introduzam mudanças significativas. Depois de AWS identificar uma versão como um produto estável, ela pode marcá-la como candidata a lançamento.

Os candidatos a lançamento estão prontos para o lançamento do GA, a menos que surjam bugs significativos, e receberão suporte total para AWS .

- Disponibilidade geral (GA) (Fase 1) - Durante essa fase, os SDKs são totalmente suportados. AWS fornecerá lançamentos regulares do SDK que incluem suporte para novos serviços, atualizações de API para serviços existentes, bem como correções de bugs e segurança. Para Ferramentas, AWS fornecerá lançamentos regulares que incluem novas atualizações de recursos e correções de erros. AWS suportará a versão GA de um SDK por pelo menos 24 meses.
- Anúncio de manutenção (Fase 2) - AWS fará um anúncio público pelo menos 6 meses antes de um SDK entrar no modo de manutenção. Durante esse período, o SDK continuará sendo totalmente suportado. Normalmente, o modo de manutenção é anunciado ao mesmo tempo em que a próxima versão principal é transferida para GA.
- Manutenção (Fase 3) - Durante o modo de manutenção, AWS limita as versões do SDK para tratar apenas de correções críticas de bugs e problemas de segurança. Um SDK não receberá atualizações de API para serviços novos ou existentes, nem será atualizado para oferecer suporte a novas regiões. O modo de manutenção tem uma duração padrão de 12 meses, a menos que especificado de outra forma.
- Fim do suporte (Fase 4) - Quando um SDK chega ao fim do suporte, ele não receberá mais atualizações ou lançamentos. As versões publicadas anteriormente continuarão disponíveis por meio de gerenciadores de pacotes públicos e o código permanecerá ativado GitHub. O GitHub repositório pode ser arquivado. O uso de um SDK alcançado end-of-support é feito a critério do usuário. Recomendamos que os usuários atualizem para a nova versão principal.

Veja a seguir uma ilustração visual do ciclo de vida da versão principal do SDK. Observe que os cronogramas mostrados abaixo são ilustrativos e não vinculativos.

## Ciclo de vida da dependência

A maioria dos AWS SDKs tem dependências subjacentes, como tempos de execução de linguagem, sistemas operacionais ou bibliotecas e estruturas de terceiros. Essas dependências geralmente estão vinculadas à comunidade linguística ou ao fornecedor que possui esse componente específico. Cada comunidade ou fornecedor publica sua própria end-of-support programação para seu produto.

Os termos a seguir são usados para classificar as dependências subjacentes de terceiros:

- Sistema operacional (SO): exemplos incluem Amazon Linux AMI, Amazon Linux 2, Windows 2008, Windows 2012, Windows 2016, etc.

- Language Runtime: exemplos incluem Java 7, Java 8, Java 11, .NET Core, .NET Standard, .NET PCL etc.
- Biblioteca/estrutura de terceiros: exemplos incluem OpenSSL, .NET Framework 4.5, Java EE etc.

Nossa política é continuar oferecendo suporte às dependências do SDK por pelo menos 6 meses após a comunidade ou o fornecedor encerrar o suporte para a dependência. Essa política, no entanto, pode variar dependendo da dependência específica.

#### Note

AWS reserva o direito de interromper o suporte para uma dependência subjacente sem aumentar a versão principal do SDK

## Métodos de comunicação

Os anúncios de manutenção são comunicados de várias maneiras:

- Um anúncio por e-mail é enviado às contas afetadas, anunciando nossos planos de encerrar o suporte para a versão específica do SDK. O e-mail descreverá o caminho end-of-support, especificará os cronogramas da campanha e fornecerá orientações de atualização.
- A documentação do SDK, como documentação de referência da API, guias do usuário, páginas de marketing de produtos do SDK e GitHub readme (s), é atualizada para indicar o cronograma da campanha e fornecer orientação sobre a atualização dos aplicativos afetados.
- É publicada uma postagem no AWS blog que descreve o caminho e reitera os cronogramas da campanha. end-of-support
- Os avisos de depreciação são adicionados aos SDKs, descrevendo o caminho end-of-support e vinculando à documentação do SDK.

Para ver a lista das principais versões disponíveis dos AWS SDKs e das ferramentas e onde elas estão em seu ciclo de vida de manutenção, consulte. [Matriz de suporte a versões](#)

## AWS Matriz de suporte de versões de SDKs e ferramentas

A matriz abaixo mostra a lista das principais versões disponíveis do AWS Software Development Kit (SDK) e onde elas estão no ciclo de vida de manutenção com os cronogramas associados. Para obter informações detalhadas sobre o ciclo de vida das principais versões dos AWS SDKs e das ferramentas e suas dependências subjacentes, consulte [Política de manutenção](#)

SDK	Versão principal	Fase atual	Data da disponibilidade geral	Observações
<a href="#">AWS CLI</a>	1.x	Disponibilidade geral	02/09/2013	
<a href="#">AWS CLI</a>	2.x	Disponibilidade geral	2/10/2020	
<a href="#">SDK for C++</a>	1.x	Disponibilidade geral	02/09/2015	
<a href="#">SDK for Go V2</a>	V2 1.x	Disponibilidade geral	19/01/2021	
<a href="#">SDK for Go</a>	1.x	Anúncio de manutenção	19/11/2015	Veja <a href="#">o anúncio</a> para obter detalhes e datas
<a href="#">SDK para Java</a>	1.x	Anúncio de manutenção	25/03/2010	Veja <a href="#">o anúncio</a> para obter detalhes e datas
<a href="#">SDK para Java</a>	2.x	Disponibilidade geral	20/11/2018	
<a href="#">SDK para JavaScript</a>	1.x	Fim do suporte	6/5/2013	

SDK	Versão principal	Fase atual	Data da disponibilidade geral	Observações
<a href="#">SDK para JavaScript</a>	2.x	Anúncio de manutenção	19/06/2014	Veja <a href="#">o anúncio</a> para obter detalhes e datas
<a href="#">SDK para JavaScript</a>	3.x	Disponibilidade geral	15/12/2020	
<a href="#">SDK for Kotlin</a>	1.x	Disponibilidade geral	27/11/2023	
<a href="#">SDK for .NET</a>	1.x	Fim do suporte	11/2009	
<a href="#">SDK for .NET</a>	2.x	Fim do suporte	08/11/2013	
<a href="#">SDK for .NET</a>	3.x	Disponibilidade geral	28/07/2015	
<a href="#">SDK for PHP</a>	2.x	Fim do suporte	02/11/2012	
<a href="#">SDK for PHP</a>	3.x	Disponibilidade geral	27/05/2015	
<a href="#">SDK for Python (Boto2)</a>	1.x	Fim do suporte	13/07/2011	
<a href="#">SDK for Python (Boto3)</a>	1.x	Disponibilidade geral	22/06/2015	
<a href="#">SDK for Python (Botocore)</a>	1.x	Disponibilidade geral	22/06/2015	
<a href="#">SDK for Ruby</a>	1.x	Fim do suporte	14/07/2011	
<a href="#">SDK for Ruby</a>	2.x	Fim do suporte	15/02/2015	



SDK	Versão principal	Fase atual	Data da disponibilidade geral	Observações
<a href="#">SDK for Ruby</a>	3.x	Disponibilidade geral	29/08/2017	
<a href="#">SDK for Rust</a>	1.x	Disponibilidade geral	27/11/2023	
<a href="#">SDK for Swift</a>	1.x	Pré-visualização para desenvolvedores		
<a href="#">Ferramentas para PowerShell</a>	2.x	Fim do suporte	08/11/2013	
<a href="#">Ferramentas para PowerShell</a>	3.x	Fim do suporte	29/07/2015	
<a href="#">Ferramentas para PowerShell</a>	4.x	Disponibilidade geral	21/11/2019	

# Guia de referência do histórico de documentos para AWS SDKs e ferramentas

A tabela a seguir descreve adições e atualizações importantes no Guia de referência de SDKs AWS e ferramentas. Para receber notificações sobre atualizações dessa documentação, você pode se inscrever em o feed RSS.

Alteração	Descrição	Data
<a href="#">Propriedades do sistema SDK for Java 1.x</a>	Adicione detalhes sobre as configurações do sistema JVM suportadas pela versão 1.x. AWS SDK for Java	30 de maio de 2024
<a href="#">Atualizações de configurações</a>	Adicione as configurações do sistema JVM.	27 de março de 2024
<a href="#">Atualizações da tabela de compatibilidade</a>	Atualizações na compatibilidade do suporte ao SDK, atualizações nos procedimentos do IAM Identity Center.	20 de fevereiro de 2024
<a href="#">Atualização da credencial do contêiner. Atualização do IMDS.</a>	Adicionando suporte para o Amazon EKS. Adicionar configuração para desativar o fallback do IMDSv1.	29 de dezembro de 2023
<a href="#">Compactação de solicitações</a>	Adicionar configurações para o recurso de compactação de solicitações.	27 de dezembro de 2023
<a href="#">Tabelas de compatibilidade</a>	Tabelas de compatibilidade para SDK e recursos de ferramentas atualizados para incluir SDK para Kotlin, SDK para Rust e AWS Tools for PowerShell.	10 de dezembro de 2023

<a href="#">Atualizações de autenticação</a>	Atualizações nos métodos de autenticação compatíveis para SDKs e ferramentas.	1º de julho de 2023
<a href="#">Atualizações de práticas recomendadas do IAM</a>	Guia atualizado para alinhamento com as práticas recomendadas do IAM. Para obter mais informações, consulte <a href="#">Práticas recomendadas de segurança no IAM</a> .	27 de fevereiro de 2023
<a href="#">Atualizações em SSO</a>	Atualizações nas credenciais de SSO para a nova configuração do token SSO.	19 de novembro de 2022
<a href="#">Atualizações de configurações</a>	Atualizações na tabela de suporte para configuração geral e para pontos de acesso multirregionais do Amazon S3.	17 de novembro de 2022
<a href="#">Atualizações de configurações</a>	Atualizações para maior clareza do cliente IMDS e das credenciais do IMDS. Atualizações nas variáveis de ambiente.	4 de novembro de 2022
<a href="#">Atualização da página de boas-vindas</a>	Anunciando a Amazon CodeWhisperer.	22 de setembro de 2022
<a href="#">Alteração do nome do serviço para login único</a>	Atualizações para refletir que o AWS SSO agora é chamado de AWS IAM Identity Center.	26 de julho de 2022
<a href="#">Atualização de configurações</a>	Pequenas atualizações nos detalhes do arquivo de configuração e nas configurações suportadas.	15 de junho de 2022

[Atualização](#)

Atualização massiva de quase todas as partes deste guia. 1º de fevereiro de 2022

[Lançamento inicial](#)

A primeira versão deste guia foi lançada ao público. 13 de março de 2020

# Glossário do AWS

Para obter a terminologia mais recente da AWS, consulte o [glossário da AWS](#) na Referência do Glossário da AWS.

As traduções são geradas por tradução automática. Em caso de conflito entre o conteúdo da tradução e da versão original em inglês, a versão em inglês prevalecerá.