AWS软件开发工具包和工具

参考指南

AWS软件开发工具包和工具: 参考指南

Table of Contents

AWSSDK 和工具参考指南

许多 SDK 和工具通过共享设计规范或共享库共享一些通用功能。

本指南包含有关以下内容的信息：

- 配置 (p. 2)— 如何使用共享config和credentials文件或环境变量来配置 AWS SDK 和工具。
- 身份验证和访问 (p. 8)— 确定您的代码或工具在使用开发AWS时如何进行身份验证。AWS 服务
- 配置和身份验证设置参考 (p. 19)— 所有可用于身份验证和配置的标准化设置的参考。
- AWS通用运行时 (CRT) 库 (p. 72)— 几乎所有 SDK 都可用的共享AWS通用运行时 (CRT) 库概述。
- AWSSDK 和工具维护政策 (p. 73)涵盖AWS软件开发套件 (SDK) 和工具（包括移动和物联网 (IoT) SDK）及其底层依赖项的维护策略和版本控制。

本 AWS SDK 和工具参考指南旨在作为适用于多个 SDK 和工具的信息库。除此处提供的任何信息外，还应使用您正在使用的 SDK 或工具的特定指南。以下是 SDK 和工具，其中包含本指南中的相关材料部分：

如果你使用的是：	本指南中与您相关的部分有：
• 任何 SDK 或工具	AWSSDK 和工具维护政策 (p. 73)
• AWS Cloud Development Kit (AWS CDK) 开发人员指南 • AWS Serverless Application Model 开发人员指南 • AWS Toolkit for Eclipse 用户指南 • AWS Toolkit for JetBrains 用户指南 • AWS Toolkit for Visual Studio 用户指南 • AWS Toolkit for Visual Studio Code 用户指南	配置 (p. 2) 身份验证和访问 (p. 8) AWSSDK 和工具维护政策 (p. 73)
• AWS Command Line Interface 用户指南 • AWS SDK for C++ 开发人员指南 • AWS SDK for Go 开发人员指南 • AWS SDK for Java 开发人员指南 • AWS SDK for JavaScript 开发人员指南 • AWS SDK for .NET 开发人员指南 • AWS SDK for PHP 开发人员指南 • AWS适用于 Python 的 SDK (Boto3) 入门 • AWS SDK for Ruby 开发人员指南 • AWS Tools for Windows PowerShell 用户指南	配置 (p. 2) 身份验证和访问 (p. 8) 配置和身份验证设置参考 (p. 19) AWS通用运行时 (CRT) 库 (p. 72) AWSSDK 和工具维护政策 (p. 73)

开发者资源

Amazon CodeWhisperer 是一项基于机器学习 (ML) 的服务，可根据集成开发环境 (IDE) 中的代码注释和代码生成代码建议，从而帮助提高开发人员的工作效率。要详细了解支持哪些语言和 IDE 以及如何注册免费预览版，请参阅亚马逊 CodeWhisperer。

配置

使用AWS软件开发工具包和其他AWS开发者工具（例如 AWS Command Line Interface (AWS CLI)），您可以与AWS服务 API 进行交互。但是，在尝试执行此操作之前，必须使用执行请求的操作所需的信息来配置 SDK 或工具。

此信息包括以下项目：

- 用于识别谁在调用 API 的@@ 凭证信息。凭据用于加密向AWS服务器发出的请求。使用此信息AWS确认您的身份，并可以检索与之相关的权限策略。然后，它可以确定允许您执行哪些操作。
- 其他配置详细信息，用于告知AWS CLI或 SDK 如何处理请求、将请求发送到何处（发送到哪个AWS服务端点）以及如何解释或显示响应。

每个 SDK 或工具都支持多个来源，您可以使用这些来源来提供所需的凭据和配置信息。有些来源是 SDK 或工具所独有的，有关如何使用该方法的详细信息，您必须参阅该工具或 SDK 的文档。

但是，大多数 AWS SDK 和工具都支持来自两个主要来源（除了代码本身）的常见设置：

- 共享AWS配置和凭据文件 (p. 2)-共享`credentials`文件`config`和文件是为 AWS SDK 或工具指定身份验证和配置的最常用方式。使用这些文件存储您的工具和应用程序可以使用的设置。共享`credentials`文件`config`和文件中的设置与特定的配置文件相关联。使用多个配置文件，您可以创建不同的设置配置以应用于不同的场景。当你使用AWS工具调用命令或使用 SDK 调用 AWS API 时，你可以指定用于该操作的配置文件以及配置设置。其中一个配置文件被指定为配置文件，`default`当您未明确指定要使用的配置文件时，该配置文件会自动使用。本参考指南中记录了您可以存储在这些文件中的设置。
- 环境变量 (p. 6)-某些设置也可以存储在操作系统的环境变量中。尽管一次只能有一组有效的环境变量，但是随着程序的运行和需求的变化，可以很容易地对其进行动态修改。

此部分中的其他主题

已共享config和credentials文件

共享的AWS `config`和`credentials`文件包含一组配置文件。配置文件是一组配置值，可以使用其配置文件名称从 SDK/工具中引用这些值。配置值附加到配置文件中，以便在使用该配置文件时配置 SDK/工具的某些方面。

通常，您可以在共享中放置的任何值`credentials`也可以将文件放在共享文件中`config`文件。反之亦然；只能将几个设置放在`credentials`文件。但是，作为安全最佳实践，我们建议您将所有敏感值（例如访问密钥 ID 和私有密钥）单独保存`credentials`文件。这样，如有必要，您可以为每个文件提供单独的权限。

两者都是共享的`config`和`credentials`文件是仅包含 ASCII 字符（UTF-8 编码）的纯文本文件。它们采用通常所说的形式INI 文件。

配置文件

共享中的设置`config`和`credentials`文件与特定的配置文件相关联。使用多个配置文件，您可以创建不同的设置配置以应用于不同的场景。

的[default]如果未指定特定的命名配置文件，profile 包含由 SDK 或工具操作使用的值。您也可以创建单独的配置文件，您可以按名称明确引用这些配置文件。每个已命名的配置文件可以具有不同的设置组。

[default]只是一个未命名的个人资料。此个人资料名为default因为如果用户未指定配置文件，则它是 SDK 使用的默认配置文件。它不为其他配置文件提供继承的默认值。例如，如果你在[default]profile 并且您没有在命名的配置文件中进行设置，则在使用命名配置文件时不会设置该值。

（可选）设置要通过 SDK 代码使用的命名配置文件或AWS CLI命令。或者，您可以使用环境变量AWS_PROFILE以指定要使用的配置文件设置。

Linux/macOS 通过命令行设置环境变量的示例：

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

Windows 通过命令行设置环境变量的示例：

```
setx AWS_PROFILE "my_default_profile_name"
```

配置文件的格式

的config文件被组织成几个部分。分区是设置的命名集合，一直持续到遇到另一个分区定义行。

的configfile 是使用以下格式的纯文本文件：

- 部分中的所有条目均采用 setting-name=value 的一般形式。
- 可以通过以井号字符 (#) 开头来注释掉行。

章节类型

分区定义是将名称应用于设置集合的行。章节定义行以方括号开头和结尾 ([ ])。方括号内有一个章节类型标识符和该部分的自定义名称。你可以使用字母、数字、连字符 (-) 和下划线 (_)，但没有空格。

章节类型：profile

章节定义行示例：[profile *dev*]

的profilesection definition line 命名了可以在不同场景中应用的配置分组。[default]是唯一不需要的配置文件profile章节标识符。要更好地了解已命名的配置文件，请参阅前面关于配置文件的部分。

以下示例显示了一个基本的示例config带有 a 的文件[default]个人资料。它设定了region (p. 41)设置。

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

以下示例显示了config带有 a 的文件profile章节定义行。它使用标识符profile后面是配置文件的唯一名称。此行之后的所有设置，直到遇到另一个分区定义为止，都将包含在此命名的配置文件中。

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

有些设置有自己的嵌套子设置组，例如s3以下示例中的设置和子设置。通过缩进一个或多个空格将子设置与群组相关联。

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

章节类型：sso-session

章节定义行示例：[sso-session *my-sso*]

的sso-session分区定义行命名一组设置，用于配置要解析的配置文件AWS凭据使用AWS IAM Identity Center。有关配置单点登录身份验证的更多信息，请参阅IAM Identity Center 身份验证 (p. 9)。个人资料已链接到sso-session按键值对分段，其中sso-session是你的钥匙和名字sso-session部分是值，例如sso-session = <name-of-sso-session-section>。

以下示例配置了短期配置文件AWS” 的凭证SampleRole“使用 “my-sso” 中的令牌在 “111122223333” 账户中扮演的 IAM 角色。“my-sso”sso-session部分引用在profile按名称分段使用sso-session钥匙。

```
[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
```

章节类型：services

章节定义行示例：[services *dev*]

> **Note**
>
> 的services部分支持服务特定的端点自定义，并且仅在包含此功能的 SDK 和工具中可用。要查看您的 SDK 是否可以使用此功能，请参阅与之兼容AWSSDKS (p. 55)用于特定服务的终端节点。

的services部分定义行命名一组设置，用于为其配置自定义终端节点AWS 服务请求。个人资料已链接到services按键值对分段，其中services是你的钥匙和名字services部分是值，例如services = <name-of-services-section>。

的services部分通过以下方式进一步分成小节<SERVICE> = 线，哪里<SERVICE>是AWS 服务标识符密钥。的AWS 服务标识符基于 API 模型的serviceId用下划线替换所有空格并将所有字母小写。有关要在中使用的所有服务标识符密钥的列表services部分，请参阅服务特定端点的标识符 (p. 56)。服务标识符密钥后面是嵌套设置，每个设置在自己的行上，缩进两个空格。

以下示例使用了services定义配置终端节点，使其仅用于向用户发出的请求Amazon DynamoDB服务。的"local-dynamodb" services部分引用在profile按名称分段使用services钥匙。的AWS 服务标识符密钥是dynamodb。的Amazon DynamoDB服务小节从线路开始dynamodb = 。紧随其后的任何缩进行都包含在该小节中，并适用于该服务。

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

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

有关自定义终端节点配置的更多信息，请参阅特定于服务的终端节点 (p. 52)。

凭证文件的格式

的规则credentials文件通常与的相同configfile，只是个人资料部分不是以单词开头profile。在方括号之间仅使用唯一的配置文件名称本身。

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

您只能将一小部分设置和值存储在credentials文件。通常，只有那些值才会被视为 “机密” 或敏感值，例如访问密钥 ID 和密钥。本指南中每项设置的页面都说明了是否可以将其存储在credentials文件或仅在config文件。

以下示例显示了一个基本的示例credentials带有 a 的文件[default]个人资料。它设定了aws_access_key_id和aws_secret_access_key (p. 35)全局设置。

```
[default]
aws_access_key_id=AKIAIOSFODNN7EXAMPLE
aws_secret_access_key=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
aws_session_token=IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZVERYLONGSTRINGEXAMPL
```

共享文件的位置config和credentials文件

共享的AWS config和credentials文件是纯文本文件，默认情况下位于名为的文件夹中.aws那放在”home“您计算机上的文件夹。

在 Linux 和 macOS 上，这通常显示为~/.aws。在 Windows 上，它是 %USERPROFILE%\.aws。

操作系统	文件的默认位置和名称
Linux 和 macOS	~/.aws/config ~/.aws/credentials
Windows	%USERPROFILE%\.aws\config %USERPROFILE%\.aws\credentials

一个~/要么~然后是文件系统的默认路径分隔符，然后是路径开头的默认路径分隔符，通过按顺序检查

1. （所有平台）HOME环境变量
2. （Windows 平台）USERPROFILE环境变量
3. （Windows 平台）HOMEDRIVE环境变量，前置于HOMEPATH环境变量（例如$HOMEDRIVE$HOMEPATH)
4. （每个 SDK 或工具可选）特定于 SDK 或工具的主路径解析函数或变量

如果可能的话，如果在路径的开头指定了用户的主目录（例如~username/），它会被解析到请求的用户名的主目录（例如/home/username/.aws/config)。

更改这些文件的默认位置：

可以设置以下环境变量，将这些文件的位置或名称从默认值更改为自定义值：

- config文件环境变量：**AWS_CONFIG_FILE**

- credentials文件环境变量：**AWS_SHARED_CREDENTIALS_FILE**

Linux/macOS

您可以通过运行以下命令来指定其他位置出口在 Linux 或 macOS 上运行命令。

```
$ 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

您可以通过运行以下命令来指定其他位置setxWindows 上的命令。

```
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
```

环境变量支持

环境变量提供了另一种指定配置选项和凭证的方法；若要编写脚本或将一个命名配置文件临时设置为默认配置文件，环境变量会很有用。有关大多数 SDK 支持的环境变量列表，请参阅环境变量列表 (p. 21)。

选项的优先顺序

- 如果您使用环境变量来指定设置，则它将覆盖从共享配置文件加载的任何值AWS config和credentials文件。
- 如果您通过使用上的参数来指定设置AWS CLI命令行，它会覆盖相应环境变量或配置文件中配置文件中的任何值。

如何设置环境变量

下面的示例介绍您如何可以为默认用户配置环境变量。

Linux, macOS, or Unix

```
$ export AWS_ACCESS_KEY_ID=AKIAIOSFODNN7EXAMPLE
$ export AWS_SECRET_ACCESS_KEY=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
$ export
 AWS_SESSION_TOKEN=AQoEXAMPLEH4aoAH0gNCAPy...truncated...zrkuWJOgQs8IZZaIv2BXIa2R4Olgk
$ export AWS_DEFAULT_REGION=us-west-2
```

设置环境变量会更改使用的值，直到 Shell 会话结束或直到您将该变量设置为其他值。通过在 shell 的启动脚本中设置变量，可使变量在未来的会话中继续有效。

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...zrkuWJOgQs8IZZaIv2BXIa2R4Olgk
C:\> setx AWS_DEFAULT_REGION us-west-2
```

使用set要设置环境变量，请更改当前命令提示符会话结束之前使用的值，或者直到您将变量设置为其他值。使用setx要设置环境变量，会更改当前命令提示符会话和运行该命令后创建的所有命令提示符会话中使用的值。它不 影响在运行该命令时已经运行的其他命令 shell。

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...zrkuWJOgQs8IZZaIv2BXIa2R4Olgk"
PS C:\> $Env:AWS_DEFAULT_REGION="us-west-2"
```

如果您将环境变量设置为PowerShellprompt 如前面的示例所示，它仅在当前会话的持续时间内保存该值。使环境变量设置在所有环境中保持不变PowerShell和命令提示符会话，请使用系统申请于控制面板。或者，您可以为所有将来设置该变量PowerShell通过将其添加到您的会话中来进行会话PowerShell个人资料。请参阅PowerShell文档了解有关存储环境变量或在会话之间保存环境变量的更多信息。

无服务器环境变量设置

如果您使用无服务器架构进行开发，则还有其他设置环境变量的选项。根据您的容器，您可以对在这些容器中运行的代码使用不同的策略来查看和访问环境变量，类似于非云环境。

例如，用AWS Lambda，你可以直接设置环境变量。有关详细信息，请参阅使用AWS Lambda环境变量在AWS Lambda开发者指南。

在无服务器框架中，您通常可以在以下位置设置 SDK 环境变量`serverless.yml`环境设置下的提供者密钥下的文件。有关信息`serverless.yml`文件，请参阅常规功能设置在无服务器框架文档中。

无论您使用哪种机制来设置容器环境变量，都有一些变量由容器保留，例如在 Lambda 上记录的变量定义的运行时环境变量。请务必查阅您使用的容器的官方文档，以确定如何处理环境变量以及是否存在任何限制。

身份验证和访问

您必须确定您的代码是如何进行身份验证的AWS当你与之一起发展时AWS 服务。您可以将编程访问权限配置为AWS以不同的方式提供资源，具体取决于环境和AWS您可以访问权限。

本地运行的代码的身份验证选项（不在AWS)

- IAM Identity Center 身份验证 (p. 9)— 作为安全最佳实践，我们建议使用AWS Organizations使用 IAM 身份中心管理所有人的访问权限AWS 账户。你可以在中创建用户AWS IAM Identity Center、使用微软 Active Directory、使用 SAML 2.0 身份提供商 (IdP)，或者单独将你的 IdP 联合到AWS 账户。要检查您所在的地区是否支持 IAM 身份中心，请参阅AWS IAM Identity Center终端节点和配额在Amazon Web Services 一般参考。
- IAM Roles Anywhere (p. 12)— 您可以使用 IAM Roles Anywhere 在 IAM 中为在外部运行的服务器、容器和应用程序等工作负载获取临时安全证书AWS。要在任何地方使用 IAM 角色，您的工作负载必须使用 X.509 证书。
- 其他身份验证方式 (p. 13)— 其他可能不太方便或可能增加您的安全风险的选项AWS资源。

在中运行的代码的身份验证选项AWS环境

- 为亚马逊 EC2 实例使用 IAM 角色 (p. 16)— 使用 IAM 角色在 Amazon EC2 实例上安全地运行您的应用程序。
- 您可以通过编程方式与之交互AWS通过以下方式使用 IAM 身份中心：
  - 使用AWS CloudShell要运行AWS CLI来自控制台的命令。
  - 使用AWS Cloud9开始编程AWS使用集成开发环境 (IDE)AWS资源。
  - 要尝试为软件开发团队提供基于云的协作空间，请考虑使用亚马逊CodeCatalyst。

有关访问管理的更多信息

的IAM 用户指南包含以下有关安全控制访问的信息AWS资源：

- IAM 身份（用户、用户组和角色）— 了解身份的基础知识AWS。
- IAM 中的安全最佳实践— 开发时要遵循的安全建议AWS根据以下条件进行申请责任共担模型。

的Amazon Web Services 一般参考具有以下基础知识：

- 了解并获得你的AWS证书— 访问控制台和编程访问的关键选项和管理实践。

AWS 构建者 ID

你的AWS 构建者 ID与任何东西互补AWS 账户您可能已经拥有或想要创建。虽然一个AWS 账户充当容器AWS您创建的资源并为这些资源提供了安全边界，您的AWS 构建者 ID代表你作为一个个体。你可以用你的账号登录AWS 构建者 ID访问开发者工具和服务，例如亚马逊CodeWhisperer和亚马逊CodeCatalyst。

- 使用登录AWS 构建者 ID在AWS 登录用户指南— 学习如何创建和使用AWS 构建者 ID并了解生成器 ID 提供的内容。
- 使用进行身份验证CodeWhisperer和AWS Toolkit-生成器 ID在CodeWhisperer用户指南— 了解如何做CodeWhisperer使用AWS 构建者 ID。

- CodeCatalyst概念-AWS 构建者 ID在亚马逊CodeCatalyst用户指南— 了解如何做CodeCatalyst使用AWS构建者 ID。

IAM Identity Center 身份验证

AWS IAM Identity Center是在非AWS计算服务上进行开发时推荐的提供AWS凭证的方法。例如，这将类似于您的本地开发环境。如果您正在使用诸如亚马逊弹性计算云 (Amazon EC2) AWS Cloud9 或 Amazon EC2之类的AWS资源进行开发，我们建议您从该服务获取证书。

使用 IAM 身份中心配置编程访问权限

步骤 1：建立访问权限并选择相应的权限集

选择下列方法之一来访问您的 AWS 凭证。

我尚未通过 IAM Identity Center 确立访问权限

请按照《AWS IAM Identity Center 用户指南》的入门中的说明操作。此过程支持 IAM Identity Center，创建管理用户并添加适当的最低权限集。

- 对于步骤 6 – 创建应用最低权限的权限集。除非您的雇主已为此目的创建自定义权限集，否则建议使用预定义的 `PowerUserAccess` 权限集。

退出门户并重新登录以查看 `Administrator` 或 `PowerUserAccess` 的 AWS 账户和选项。在使用开发工具包时选择 `PowerUserAccess`。这也可以帮助您找到有关编程访问的详细信息。

我已通过我的雇主管理的联邦身份提供商（例如 Azure AD 或 Okta）获得对 AWS 的访问权限

通过身份提供商的门户登录 AWS。如果您的云管理员已向您授予 `PowerUserAccess`（开发人员）权限，您将看到您有权访问的 AWS 账户和您的权限集。在您的权限集名称旁边，可以看到有关使用该权限集手动或以编程方式访问账户的选项。

自定义实现可能会产生不同的体验，例如不同的权限集名称。如果您不确定要使用哪个权限集，请联系 IT 团队以寻求帮助。

我已通过雇主管理的 AWS 访问门户获得对 AWS 的访问权限

通过 AWS 访问门户登录到 AWS。如果您的云管理员已向您授予 `PowerUserAccess`（开发人员）权限，您将看到您有权访问的 AWS 账户和您的权限集。在您的权限集名称旁边，可以看到有关使用该权限集手动或以编程方式访问账户的选项。

我已通过我的雇主管理的联合自定义身份提供商获得对 AWS 的访问权限

请联系您的 IT 团队以寻求帮助。

步骤 2：配置软件开发工具包和工具以使用 IAM 身份中心

1. 在开发计算机上，安装最新的AWS CLI。请参阅AWS Command Line Interface用户指南AWS CLI中的安装或更新最新版本的。您可以使用AWS CLI来配置和登录 IAM 身份中心访问门户。
2. （可选）要验证是否AWS CLI正在运行，请打开命令提示符并运行该`aws --version`命令。
3. 在AWS访问门户中，选择用于开发的权限集，然后选择命令行或编程访问链接。
4. 在 “获取凭据” 对话框中，根据您的操作系统选择 macOS 和 Linux 或 Windows。
5. 选择 IAM Identity Center 证书方法以获取下一步所需的`SSO Start URL`和`SSO Region`值。

6. 按照程序运行`aws configure sso`命令，使用向`aws configure sso`导配置您的配置文件。AWS CLI出现提示时，输入您在上一步中收集的配置值。

   a. 对于 CLI 配置文件名称，我们建议您在开始时输入`###`。有关如何设置非默认（已命名）配置文件及其关联环境变量的信息，请参阅配置文件 (p. 2)。

   b. 该`aws configure sso`命令还会让您登录 IAM Identity Center 访问门户并开始访问门户会话。

   c. 或者，您可以手动创建此配置。有关`config`文件设置，请参阅IAM 身份中心凭证提供商 (p. 32)。

7. （可选）在中AWS CLI，通过运行`aws sts get-caller-identity`命令确认活动会话身份。响应应显示您配置的 IAM 身份中心权限集。

8. 如果您使用的是 S AWS DK，请在您的开发环境中为 SDK 创建应用程序。

   a. 对于某些软件开发工具包，在使用 IAM Identity Center 身份验证之前，`SSOOIDC`必须将其他软件包（例如`SSO`和）添加到您的应用程序中。有关详细信息，请参阅您的特定 SDK。

   b. 如果您之前配置了对的访问权限AWS，请查看您的共享AWS`credentials`文件是否有任何访问权限AWS 访问密钥 (p. 35)。由于凭证提供者链 (p. 23)优先顺序，在软件开发工具包或工具使用 IAM Identity Center 证书之前，您必须移除所有静态证书。

要深入了解 SDK 和工具如何使用此配置使用和刷新凭据，请参阅了解 IAM 身份中心身份验证 (p. 10)。根据您配置的会话时长，您的访问权限最终将过期，SDK 或工具将遇到身份验证错误。

要在需要时再次刷新访问门户会话，AWS CLI请使用运行`aws sso login`命令。使用预定义的`PowerUserAccess`权限集，或者您或您的雇主创建的任何权限集，以最好地应用最低权限权限。

您可以延长 IAM Identity Center 访问门户会话持续时间和权限集会话持续时间。这会延长您在需要再次手动登录之前运行代码的时间。AWS CLI有关更多信息，请参阅 AWS IAM Identity Center 用户指南 中的以下主题：

- IAM Identity Center 会话持续时间 — 配置用户AWS访问门户会话的持续时间
- 权限集会话持续时间-设置会话持续时间

了解 IAM 身份中心身份验证

相关的 IAM 身份中心条款

以下术语可帮助您了解背后的流程和配置AWS IAM Identity Center。对于其中一些身份验证概念，AWSSDK API 的文档使用的名称与 IAM Identity Center 不同。知道两个名字会很有帮助。

下表显示了备用名称之间的关系。

IAM 身份中心名称	SDK API 名称	描述
身份中心	`sso`	尽管已重命名AWS单点登录，但出于向后兼容目的，`sso`API 命名空间仍将保留其原始名称。有关更多信息，请参阅AWS IAM Identity Center用户指南中的 IAM 身份中心重命名。
IAM 身份中心控制台 管理控制台		用于配置单点登录的控制台。

IAM 身份中心名称	SDK API 名称	描述
AWS访问门户网址		您的 IAM 身份中心账户独有的网址，例如`https://xxx.awsapps.com/start`。您使用 IAM 身份中心登录凭证登录此门户。
IAM 身份中心访问门户会话	身份验证会话	向呼叫者提供不记名访问令牌。
权限集会话		软件开发工具包内部用于进行调用的 IAM 会AWS 服务话。在非正式讨论中，你可能会看到它被错误地称为“角色会话”。
权限集凭证	AWS凭证 sigv4 凭证	SDK 实际用于大多数AWS 服务调用（特别是所有 sigv4 AWS 服务 调用）的凭证。在非正式讨论中，你可能会看到它被错误地称为“角色证书”。
IAM 身份中心凭证提供商	SSO 凭证提供商	如何获取证书，例如提供功能的类或模块。

了解 SDK 凭据解析 AWS 服务

IAM 身份中心 API 将持有者令牌凭证交换为 sigv4 证书。大多数AWS 服务都是 sigv4 API，但也有一些例外，比如Amazon CodeWhisperer和。Amazon CodeCatalyst以下内容描述了通过AWS IAM Identity Center支持大多数应用程序代码AWS 服务调用的凭证解析流程。

启动AWS访问门户会话

- 使用您的凭据登录会话开始该过程。
  - 使用 AWS Command Line Interface (AWS CLI) 中的`aws sso login`命令。如果您还没有活动会话，这将启动一个新的 IAM 身份中心会话。
- 启动新会话时，您将收到来自 IAM Identity Center 的刷新令牌和访问令牌。AWS CLI还会使用新的访问令牌和刷新令牌更新 SSO 缓存 JSON 文件，并使其可供软件开发工具包使用。
- 如果您已经有一个活动会话，则该AWS CLI命令将重复使用现有会话，并且将在现有会话过期时过期。要了解如何设置 IAM Identity Center 会话的时长，请参阅用户指南中的配置用户AWS访问门户会话的AWS IAM Identity Center持续时间。
  - 最大会话时长已延长至 90 天，以减少频繁登录的需求。

SDK 如何获取AWS 服务通话凭证

当您为每个服务实例化客户端对象AWS 服务时，SDK 提供访问权限。将共享AWS`config`文件的选定配置文件配置为 IAM Identity Center 凭证解析时，将使用 IAM Identity Center 来解析应用程序的证书。

- 在创建客户端时，凭证解析过程将在运行时完成。

要使用 IAM 身份中心单点登录检索 sigv4 API 的证书，软件开发工具包使用 IAM 身份中心访问令牌获取 IAM 会话。此 IAM 会话称为权限集会话，它通过担任 IAM 角色提供对软件开发工具包的AWS访问权限。

- 权限集会话持续时间与 IAM Identity Center 会话持续时间是独立设置的。

  - 要了解如何设置权限集会话持续时间，请参阅AWS IAM Identity Center用户指南中的设置会话持续时间。
- 请注意，在大多数 S AWS DK API 文档中，权限集AWS凭据也被称为凭证和 sigv4 凭证。

对软件开发工具包的 IAM Identity Center API getRoleCredentials的调用会返回权限集证书。软件开发工具包的客户端对象使用该代入的 IAM 角色来调用AWS 服务，例如让 Amazon S3 列出您账户中的存储桶。在权限集会话到期之前，客户端对象可以使用这些权限集凭据继续运行。

会话过期和刷新

使用时SSO 令牌提供商配置 (p. 32)，将使用刷新令牌自动刷新从 IAM Identity Center 获取的每小时访问令牌。

- 如果访问令牌在 SDK 尝试使用时已过期，SDK 将使用刷新令牌尝试获取新的访问令牌。IAM 身份中心会将刷新令牌与您的 IAM 身份中心访问门户会话持续时间进行比较。如果刷新令牌未过期，IAM Identity Center 将使用另一个访问令牌进行响应。
- 此访问令牌可用于刷新现有客户端的权限集会话，也可以用于解析新客户端的凭据。

但是，如果 IAM Identity Center 访问门户会话已过期，则不会授予新的访问令牌。因此，无法续订权限集持续时间。每当现有客户端的缓存权限集会话长度超时时，它就会过期（并且访问权限将丢失）。

在 IAM Identity Center 会话到期后，任何创建新客户端的代码都将无法通过身份验证。这是因为未缓存权限集凭证。在您拥有有效的访问令牌之前，您的代码将无法创建新客户端并完成凭证解析过程。

总而言之，当 SDK 需要新的权限集凭证时，SDK 会首先检查所有有效的现有凭证并使用这些凭证。无论证书是针对新客户端，还是证书已过期的现有客户端，这都适用。如果找不到证书或证书无效，则软件开发工具包会调用 IAM Identity Center API 来获取新证书。要调用 API，它需要访问令牌。如果访问令牌已过期，软件开发工具包将使用刷新令牌尝试从 IAM Identity Center 服务获取新的访问令牌。如果您的 IAM 身份中心访问门户会话未过期，则会授予此令牌。

IAM Roles Anywhere

您可以使用 IAM Roles Anywhere 在 IAM 中为在外部运行的服务器、容器和应用程序等工作负载获取临时安全证书AWS。要在任何地方使用 IAM 角色，您的工作负载必须使用 X.509 证书。您的云管理员应提供所需的证书和私钥，以便将 IAM 角色配置为凭证提供商 Anywhere。

步骤 1：随时随地配置 IAM 角色

IAM Roles Anywhere 提供了一种获取在外部运行的工作负载或流程的临时证书的方法AWS。与证书颁发机构建立信任锚以获取关联的 IAM 角色的临时证书。该角色设置当您的代码通过 IAM 角色进行身份验证时，您的工作负载将拥有的权限。

有关设置信任锚点、IAM 角色和 IAM 角色随处可见配置文件的步骤，请参阅在中创建信任锚和个人资料AWS Identity and Access Management角色随处可见在无处不在 IAM 角色用户指南。

> Note
>
> 一个轮廓在无处不在 IAM 角色用户指南指的是 IAM Roles Anywhere 服务中的一个独特概念。它与共享中的个人资料无关AWS `config`文件。

第 2 步：随时随地使用 IAM 角色

要从 IAM Roles Anywhere 获取临时安全证书，请使用 IAM Roles Anywhere 提供的凭证帮助工具。凭证工具可实现 IAM 角色的签名流程 Anywhere。

有关下载凭证帮助工具的说明，请参阅从获取临时安全证书AWS Identity and Access Management角色随处可见在无处不在 IAM 角色用户指南。

使用来自 IAM 角色的临时安全证书 AnywhereAWS软件开发工具包和AWS CLI，你可以配置`credential_process`在共享中设置AWS `config`文件。软件开发工具包和AWS CLI支持使用流程凭证提供商`credential_process`进行身份验证。下面显示了要设置的一般结构`credential_process`。

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

的`credential-process`helper 工具的命令会返回与兼容的标准 JSON 格式的临时证书`credential_process`设置。请注意，命令名称包含连字符，但设置名称包含下划线。该命令需要以下参数：

- `private-key`— 签署请求的私钥的路径。
- `certificate`— 证书的路径。
- `role-arn`— 要为其获取临时证书的角色的 ARN。
- `profile-arn`— 为指定角色提供映射的配置文件的 ARN。
- `trust-anchor-arn`— 用于进行身份验证的信任锚的 ARN。

您的云管理员应提供证书和私钥。所有三个 ARN 值都可以从中复制AWS Management Console。以下示例显示了共享的`config`配置从帮助工具检索临时凭证的文件。

```
[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
```

有关可选参数和其他帮助工具的详细信息，请参阅IAM 角色随处可见凭证助手上GitHub。

有关 SDK 配置设置本身和流程凭证提供程序的详细信息，请参阅流程凭证提供商 (p. 30)在本指南中。

其他身份验证方式

使用短期证书

我们建议您配置要使用的 SDK 或工具IAM Identity Center 身份验证 (p. 9)使用延长会话持续时间选项。

但是，要直接设置 SDK 或工具的临时证书，请参阅使用短期凭证进行身份验证 (p. 14)。

使用长期证书

> Warning
>
> 为了避免安全风险，在开发专用软件或处理真实数据时，请勿使用 IAM 用户进行身份验证。而是使用与身份提供商的联合身份验证，例如 AWS IAM Identity Center。

管理跨区域的访问权限AWS 账户

作为安全最佳实践，我们建议使用AWS Organizations使用 IAM 身份中心管理所有人的访问权限AWS 账户。有关更多信息，请参阅《IAM 用户指南》中的 IAM 安全最佳实践。

你可以在 IAM Identity Center 中创建用户，使用微软 Active Directory，使用 SAML 2.0 身份提供商 (IdP)，或者单独将你的 IdP 联合到AWS 账户。使用其中一种方法，您可以为用户提供单点登录体验。您还可以强制执行多因素身份验证 (MFA)，并将临时证书用于AWS 账户访问。这与 IAM 用户不同，后者是一种可以共享的长期证书，可能会增加您的安全风险AWS资源。

仅为沙盒环境创建 IAM 用户

如果你是新手AWS，然后你可以创建一个 IAM 用户然后用它来运行教程并探索什么AWS必须提供。在学习时可以使用这种凭证，但我们建议您避免在沙盒环境之外使用。

对于以下用例，开始使用 IAM 用户可能是有意义的AWS:

- 开始使用你的AWSSDK 或工具并进行探索AWS 服务在沙盒环境中。
- 作为学习的一部分，运行计划脚本、作业和其他不支持人工参与登录过程的自动化流程。

如果您在这些用例之外使用 IAM 用户，请过渡到 IAM Identity Center 或将您的身份提供商联合到AWS 账户尽快。有关更多信息，请参阅中的身份联合AWS。

安全 IAM 用户访问密钥

您应该定期轮换 IAM 用户访问密钥。按照中的指导进行操作轮换访问密钥在IAM 用户指南。如果您认为自己不小心共享了您的 IAM 用户访问密钥，请轮换您的访问密钥。

IAM 用户访问密钥应存储在共享的AWS `credentials`本地计算机上的文件。请勿将 IAM 用户访问密钥存储在您的代码中。请勿在任何源代码管理软件中包含您的 IAM 用户访问密钥的配置文件。外部工具，例如开源项目git-秘密，可以帮助您避免无意中将敏感信息提交到 Git 存储库。有关更多信息，请参阅IAM 身份（用户、用户组和角色）在IAM 用户指南。

要设置 IAM 用户以开始使用，请参阅 使用长期凭证进行身份验证 (p. 15)。

使用短期凭证进行身份验证

我们建议您配置要使用的 SDK 或工具IAM Identity Center 身份验证 (p. 9)具有延长会话持续时间选项。但是，您可以复制和使用 AWS 访问门户中提供的临时凭证。在这些临时凭证过期后，将需要复制新凭证。您可以在配置文件中使用临时凭证，也可以将其用作系统属性和环境变量的值。

使用从中检索到的短期证书设置凭证文件AWS访问门户

1. 创建共享凭据文件。
2. 在凭证文件中，粘贴以下占位符文本，直到粘贴有效的临时凭证。

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

3. 保存该文件。该文件`~/.aws/credentials`现在应该存在于您的本地开发系统中。此文件包含[默认]个人资料如果未指定特定的命名配置文件，则 SDK 或工具将使用该配置文件。
4. 登录到 AWS 访问门户。
5. 请按照以下说明进行操作手动刷新凭据从中复制 IAM 角色证书AWS访问门户。
   a. 对于链接说明中的步骤 4，请选择授予您的开发需求访问权限的 IAM 角色名称。这个角色的名字通常像PowerUserAccess要么开发者。
   b. 对于链接说明中的第 7 步，请选择手动将个人资料添加到您的AWS凭证文件选项并复制内容。
6. 将复制的凭证粘贴到您的本地`credentials`文件。如果您使用的是，则不需要生成的配置文件名称`default`个人资料。您的文件应类似于以下内容。

```
[default]
aws_access_key_id=AKIAIOSFODNN7EXAMPLE
aws_secret_access_key=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
aws_session_token=IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZVERYLONGSTRINGEX
```

7. 保存 `credentials` 文件。

当 SDK 创建服务客户端时，它将访问这些临时凭证并将它们用于每个请求。在步骤 5a 中选择的 IAM 角色的设置决定了临时凭证的有效时间。最长持续时间为 12 小时。

在临时凭证过期后，重复步骤 4 到 7。

使用长期凭证进行身份验证

> Warning
>
> 为了避免安全风险，在开发专用软件或处理真实数据时，请勿使用 IAM 用户进行身份验证。而是使用与身份提供商的联合身份验证，例如 AWS IAM Identity Center。

如果您使用 IAM 用户运行您的代码，则开发环境中的软件开发工具包或工具将使用共享`AWScredentials`文件中的长期 IAM 用户证书进行身份验证。查看 IAM 主题中的安全最佳实践，并尽快过渡到 IAM 身份中心或其他临时证书。

有关证书的重要警告和指南

凭证警告

- 请勿使用您账户的根凭证访问AWS资源。这些凭证可提供不受限的账户访问且难以撤销。
- 请勿在应用程序文件中输入字面访问密钥或凭据信息。如果您这样做，则在将项目上传到公共存储库或在其他情况下，会有意外暴露凭证的风险。
- 不要在项目区域中包含包含凭据的文件。
- 请注意，存储在共享`AWScredentials`文件中的所有凭据都以纯文本形式存储。

有关安全管理凭证的更多指南

有关如何安全管理AWS证书的一般性讨论，请参阅中的管理AWS访问密钥的最佳实践AWS 一般参考。除了讨论之外，还要考虑以下几点：

- 使用 IAM 角色执行亚马逊弹性容器服务 (Amazon ECS) 任务的任务。
- 对在 Amazon EC2 实例上运行的应用程序使用 IAM 角色。

先决条件：创建AWS账户

要使用 IAM 用户访问AWS服务，您需要一个AWS账户和AWS证书。

1. 创建一个账户。

   要创建AWS账户，请参阅入门：你是首次AWS使用吗？ 在《AWS Account Management参考指南》中。
2. 创建管理用户。

   避免使用您的 root 用户帐户（您创建的初始帐户）来访问管理控制台和服务。相反，请按照 IAM 用户指南中的创建管理用户中所述创建一个管理用户账户。

   创建管理用户帐户并记录登录详细信息后，请务必注销根用户帐户并使用管理帐户重新登录。

这两个帐户都不适合在上面进行开发AWS或在上运行应用程序AWS。作为最佳实践，您需要创建适合这些任务的用户、权限集或服务角色。有关更多信息，请参阅《IAM 用户指南》中的应用最低权限许可。

步骤 1：创建您的 IAM 用户

- 按照《IAM 用户指南》中的创建 IAM 用户（控制台）过程操作来创建 IAM 用户。
  - 对于权限选项，选择直接附加策略以了解如何向该用户分配权限。
    - 大多数“入门”开发工具包教程都使用 Amazon S3 服务作为示例。要向应用程序提供对 Amazon S3 的完全访问权限，请选择要附加到此用户的 `AmazonS3FullAccess` 策略。
  - 您可以忽略该过程的可选步骤。

步骤 2：获取您的访问密钥

1. 在 IAM 控制台的导航窗格中，选择用户，然后选择您之前创建用户的 **`User name`**。
2. 在用户的页面上，选择安全凭证页面。然后，在访问密钥下，选择创建访问密钥。
3. 对于创建访问密钥步骤 1，选择命令行界面 (CLI) 或本地代码。这两个选项生成的密钥类型相同，可与AWS CLI和软件开发工具包一起使用。
4. 对于创建访问密钥步骤 2，输入可选标记并选择下一步。
5. 对于创建访问密钥步骤 3，选择下载.csv 文件以保存包含您的 IAM 用户访问密钥和秘密访问密钥的 `.csv` 文件。稍后您将需要此信息。

   Warning

   使用适当的安全措施来保护这些凭证的安全。
6. 选择 Done (完成)。

步骤 3：更新共享`credentials`文件

1. 创建或打开共享 AWS `credentials` 文件。此文件在 Linux 和 macOS 系统上为 `~/.aws/credentials`，在 Windows 上为 `%USERPROFILE%\.aws\credentials`。有关更多信息，请参阅凭证文件的位置。
2. 将以下文本添加到共享 `credentials` 文件中。将示例 ID 值和示例密钥值替换为您之前下载`.csv`的文件中的值。

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

3. 保存该文件。

共享`credentials`文件是存储凭据的最常用方式。它们也可以设置为环境变量，有关环境变量的名称，AWS 访问密钥 (p. 35)请参阅。这是一种入门方法，但我们建议您尽快过渡到 IAM Identity Center 或其他临时证书。停止使用长期凭证后，请记得从共享`credentials`文件中删除这些凭证。

为亚马逊 EC2 实例使用 IAM 角色

此示例介绍如何设置AWS Identity and Access Management角色具有 Amazon S3 访问权限，可在部署到 Amazon EC2 实例的应用程序中使用。

对于亚马逊弹性计算云实例，创建一个 IAM 角色，然后授予您的 Amazon EC2 实例访问该角色的权限。有关更多信息，请参阅亚马逊 EC2 的 IAM 角色在适用于 Linux 实例的亚马逊 EC2 用户指南要么亚马逊 EC2 的 IAM 角色在适用于 Windows 实例的亚马逊 EC2 用户指南。

创建 IAM 角色

创建一个 IAM 角色以授予对 Amazon S3 的只读访问权限。

1. 登录AWS Management Console，然后通过以下网址打开 IAM 控制台：https://console.aws.amazon.com/iam/。
2. 在导航窗格中，选择角色，然后选择创建角色。
3. 对于选择可信实体，下面可信实体类型，选择AWS 服务。
4. 下方用例，选择亚马逊 EC2，然后选择下一步。
5. 对于添加权限，选中复选框亚马逊 S3 只读访问权限从策略列表中，然后选择下一步。
6. 输入角色的名称，然后选择创建角色。请记住这个名字，因为当你启动 Amazon EC2 实例时，你会需要这个名字。

启动亚马逊 EC2 实例并指定您的 IAM 角色

您可以使用亚马逊 EC2 控制台启动带有 IAM 角色的亚马逊 EC2 实例。

按照中的说明启动实例适用于 Linux 实例的亚马逊 EC2 用户指南或者适用于 Windows 实例的亚马逊 EC2 用户指南。

到达核查实例启动页面时，选择编辑实例详细信息。在IAM 角色，选择您之前创建的 IAM 角色。按指示完成该过程。

> Note
>
> 您需要创建或使用现有安全组和密钥对才能连接到该实例。

通过此 IAM 和 Amazon EC2 设置，您可以将您的应用程序部署到亚马逊 EC2 实例，该实例将拥有对 Amazon S3 服务的读取权限。

连接到 EC2 实例

连接到 EC2 实例，以便您可以将示例应用程序传输到该实例，然后运行该应用程序。您需要包含用于启动实例的密钥对的私有部分的文件，即 PEM 文件。

您可以按照中的连接步骤执行此操作适用于 Linux 实例的亚马逊 EC2 用户指南或者适用于 Windows 实例的亚马逊 EC2 用户指南。当您连接时，请确保您可以将文件从开发计算机传输到您的实例。

如果你使用的是AWSToolkit，您通常也可以使用工具包连接到实例。有关更多信息，请参阅您所使用的 Toolkit 的特定用户指南。

在 EC2 实例上运行示例应用程序

1. 将应用程序文件从本地驱动器复制到您的实例。

   有关如何将文件传输到您的实例的信息，请参阅适用于 Linux 实例的亚马逊 EC2 用户指南或者适用于 Windows 实例的亚马逊 EC2 用户指南。
2. 启动应用程序并验证其运行结果是否与开发计算机上的结果相同。
3. （可选）验证应用程序是否使用 IAM 角色提供的证书。

a. 登录到 AWS Management Console 并打开 Amazon EC2 控制台（https://console.aws.amazon.com/ec2/）。
b. 选择实例并通过分离 IAM 角色行动，实例设置，附加/替换 IAM 角色。
c. 再次运行该应用程序并确认它返回了授权错误。

配置和身份验证设置参考

- 标准化凭证提供商 (p. 22)— 通用凭证提供商在多个 SDK 中实现标准化。
- 标准化功能 (p. 37)— 在多个 SDK 中实现通用功能的标准化。

软件开发工具包为提供特定于语言的 API。AWS 服务他们负责成功进行 API 调用所需的一些繁重工作，包括身份验证、重试行为等。为此，软件开发工具包采用了灵活的策略来获取用于您的请求的凭证，维护每项服务使用的设置，以及获取用于全局设置的值。

创建服务客户端

为了以编程方式进行访问AWS 服务，SDK 为每个开发工具包使用一个客户端类/对象。AWS 服务例如，如果您的应用程序需要访问 Amazon EC2，则您的应用程序会创建一个 Amazon EC2 客户端对象来与该服务接口。然后，您可以使用服务客户端向其发出请求AWS 服务。服务客户端对象是不可变的，因此您必须为向其发出请求的每个服务创建一个新的客户端，并使用不同的配置向同一服务发出请求。

设置的优先级

全局设置配置了大多数 SDK 支持并具有广泛影响的功能、凭证提供程序和其他功能。AWS 服务所有 SDK 都有一系列地点（或来源），它们会检查这些地点（或来源），以便找到全局设置的值。以下是设置查找优先级的方法：

1. 在代码中或服务客户端本身上设置的任何显式设置都优先于其他任何设置。
   - 有些设置可以根据每个操作进行设置，也可以根据需要为调用的每个操作进行更改。对于AWS CLI或 AWS Tools for PowerShell，它们采用您在命令行上输入的每个操作参数的形式。对于 SDK，显式分配可以采用您在实例化AWS 服务客户端或配置对象时或有时在调用单个 API 时设置的参数的形式。
2. 仅限 Java/Kotlin：有时会有一个 JVM 系统属性与该设置相关联。如果设置了该值，则使用该值来配置客户端。
3. 系统会检查 环境变量。如果设置了该值，则使用该值来配置客户端。
4. SDK 先检查共享`credentials`文件，然后检查共享`config`文件。如果存在该设置，则 SDK 将使用该设置。`AWS_PROFILE`环境变量或`aws.profile`系统属性可用于指定 SDK 加载哪个配置文件。
5. 最后使用 SDK 代码库本身提供的任何默认值。

> **Note**
>
> 如果`config`文件和文件中都存在相同配置`credentials`文件的设置，则使用`credentials`文件中的值而不是`config`文件中的值。

> **Note**
>
> 某些 SDK 和工具的签入顺序可能有所不同。此外，某些 SDK 和工具还支持其他存储和检索参数的方法。例如，AWS SDK for .NET支持名为 SDK Stor e 的其他来源。有关 SDK 或工具独有的提供商的更多信息，请参阅您正在使用的 SDK 或工具的特定指南。

顺序决定哪些方法优先，哪些方法会优先于其他方法。例如，如果您在共享`config`文件中设置了配置文件，则只有在 SDK 或工具先检查其他位置之后，才能找到并使用该配置文件。这意味着，如果您在`credentials`文件中放置了设置，则会使用该设置而不是`config`文件中的设置。如果您使用设置和值配置环境变量，它将覆盖`credentials`和`config`文件中的该设置。最后，单个操作（AWS CLI命令行参数或 API 参数）或代码中的设置将覆盖该命令的所有其他值。

Config文件设置列表

下表中列出的设置可以在共享AWSconfig文件中分配。它们是全球性的，影响到所有人AWS 服务。

设置名称	详细信息	
api_versions	常规配置设置 (p. 46)	
aws_access_key_id	AWS访问密钥 (p. 35)	
aws_secret_access_key	AWS访问密钥 (p. 35)	
aws_session_token	AWS访问密钥 (p. 35)	
ca_bundle	常规配置设置 (p. 46)	
credential_process	流程凭证提供商 (p. 30)	
credential_source	扮演角色凭证提供者 (p. 23)	
defaults_mode	智能配置默认值 (p. 69)	
duration_seconds	扮演角色凭证提供者 (p. 23)	
ec2_metadata_service_endpoint	IMDS 凭证提供商 (p. 28)	
ec2_metadata_service_endpoint_mode	IMDS 凭证提供商 (p. 28)	
endpoint_discovery_enabled	端点发现 (p. 45)	
endpoint_url	特定于服务的终端节点 (p. 52)	
external_id	扮演角色凭证提供者 (p. 23)	
ignore_configured_endpoint_urls	特定于服务的终端节点 (p. 52)	
max_attempts	重试行为 (p. 50)	
metadata_service_num_attempts	亚马逊 EC2 实例元数据 (p. 37)	
metadata_service_timeout	亚马逊 EC2 实例元数据 (p. 37)	
mfa_serial	扮演角色凭证提供者 (p. 23)	
output	常规配置设置 (p. 46)	
parameter_validation	常规配置设置 (p. 46)	
region	AWS 区域 (p. 41)	
retry_mode	重试行为 (p. 50)	
role_arn	扮演角色凭证提供者 (p. 23)	
role_session_name	扮演角色凭证提供者 (p. 23)	
s3_disable_multiregion_access_points	Amazon S3 多区域访问点 (p. 40)	
s3_use_arn_region	Amazon S3 接入点 (p. 39)	
source_profile	扮演角色凭证提供者 (p. 23)	

设置名称	详细信息	
sso_account_id	IAM 身份中心凭证提供商 (p. 32)	
sso_region	IAM 身份中心凭证提供商 (p. 32)	
sso_registration_scopes	IAM 身份中心凭证提供商 (p. 32)	
sso_role_name	IAM 身份中心凭证提供商 (p. 32)	
sso_start_url	IAM 身份中心凭证提供商 (p. 32)	
sts_regional_endpoints	AWS STS区域化终端节点 (p. 43)	
use_dualstack_endpoint	双栈和 FIPS 端点 (p. 44)	
use_fips_endpoint	双栈和 FIPS 端点 (p. 44)	
web_identity_token_file	担当角色凭证提供者 (p. 23)	

Credentials文件设置列表

下表中列出的设置可以在共享AWScredentials文件中分配。它们是全球性的，影响到所有人AWS 服务。

设置名称	详细信息	
aws_access_key_id	AWS访问密钥 (p. 35)	
aws_secret_access_key	AWS访问密钥 (p. 35)	
aws_session_token	AWS访问密钥 (p. 35)	

环境变量列表

下表列出了大多数 SDK 支持的环境变量。它们是全球性的，影响到所有人AWS 服务。

设置名称	详细信息	
AWS_ACCESS_KEY_ID	AWS访问密钥 (p. 35)	
AWS_CA_BUNDLE	常规配置设置 (p. 46)	
AWS_CONFIG_FILE	共享credentials文件config和文件的位置 (p. 5)	
AWS_CONTAINER_AUTHORIZATION_TOKEN	容器凭证提供商 (p. 26)	
AWS_CONTAINER_CREDENTIALS_FULL_URI	容器凭证提供商 (p. 26)	
AWS_CONTAINER_CREDENTIALS_RELATIVE_URI	容器凭证提供商 (p. 26)	
AWS_DEFAULTS_MODE	智能配置默认值 (p. 69)	
AWS_EC2_METADATA_DISABLED	IMDS凭证提供商 (p. 28)	
AWS_EC2_METADATA_SERVICE_ENDPOINT	IMDS凭证提供商 (p. 28)	

设置名称	详细信息	
AWS_EC2_METADATA_SERVICE_ENDPOINT_MODE	IMDS 凭证提供商 (p. 28)	
AWS_ENABLE_ENDPOINT_DISCOVERY	端点发现 (p. 45)	
AWS_ENDPOINT_URL	特定于服务的终端节点 (p. 52)	
AWS_ENDPOINT_URL_<SERVICE>	特定于服务的终端节点 (p. 52)	
AWS_IGNORE_CONFIGURED_ENDPOINT_URLS	特定于服务的终端节点 (p. 52)	
AWS_MAX_ATTEMPTS	重试行为 (p. 50)	
AWS_METADATA_SERVICE_NUM_ATTEMPTS	亚马逊 EC2 实例元数据 (p. 37)	
AWS_METADATA_SERVICE_TIMEOUT	亚马逊 EC2 实例元数据 (p. 37)	
AWS_PROFILE	共享config和credentials文件 (p. 2)	
AWS_REGION	AWS 区域 (p. 41)	
AWS_RETRY_MODE	重试行为 (p. 50)	
AWS_S3_DISABLE_MULTIREGION_ACCESS_POINTS	Amazon S3 多区域接入点 (p. 40)	
AWS_S3_USE_ARN_REGION	Amazon S3 接入点 (p. 39)	
AWS_SECRET_ACCESS_KEY	AWS访问密钥 (p. 35)	
AWS_SESSION_TOKEN	AWS访问密钥 (p. 35)	
AWS_SHARED_CREDENTIALS_FILE	共享credentials文件config和文件的位置 (p. 5)	
AWS_STS_REGIONAL_ENDPOINTS	AWS STS区域化终端节点 (p. 43)	
AWS_USE_DUALSTACK_ENDPOINT	双栈和 FIPS 端点 (p. 44)	
AWS_USE_FIPS_ENDPOINT	双栈和 FIPS 端点 (p. 44)	

标准化凭证提供商

许多凭证提供程序已标准化为一致的默认值，并且在许多 SDK 中以相同的方式工作。当跨多个 SDK 进行编码时，这种一致性可以提高工作效率和清晰度。所有设置都可以在代码中被覆盖。有关详细信息，请参阅您的特定 SDK API。

> **Important**
>
> 并非所有 SDK 都支持所有提供商，甚至支持提供商内部的所有方面。

主题

凭证提供者链

所有 SDK 都有一系列地点（或来源）供他们检查，以便找到用于向某人发出请求的有效凭证AWS 服务。找到有效凭证后，搜索即告停止。这种系统搜索被称为默认凭证提供者链。尽管每个 SDK 使用的不同链各不相同，但它们通常包括以下来源：

- 静态凭证（例如AWS_ACCESS_KEY_ID)。有关更多信息，请参阅AWS 访问密钥 (p. 35)：
- 来自 Web 身份令牌AWS Security Token Service(AWS STS)。有关更多信息，请参阅扮演角色凭证提供者 (p. 23)。
- AWS IAM Identity Center（有关更多信息，请参阅 IAM 身份中心凭证提供商 (p. 32)。）
- 可信实体提供商（例如AWS_ROLE_ARN)。有关更多信息，请参阅扮演角色凭证提供者 (p. 23)：
- 亚马逊弹性容器服务 (亚马逊 ECS) 证书。有关更多信息，请参阅容器凭证提供商 (p. 26)：
- 自定义凭证提供商。有关更多信息，请参阅流程凭证提供商 (p. 30)：
- 亚马逊弹性计算云 (亚马逊 EC2) 实例配置文件证书（IMDS 凭证提供商）。有关更多信息，请参阅IMDS凭证提供商 (p. 28)：

对于链中的每个步骤，都有多种分配设置值的方法。代码中指定的设置值始终优先，但也有环境变量 (p. 6)还有已共享config和credentials文件 (p. 2)。有关更多信息，请参阅 设置的优先级 (p. 19)。

扮演角色凭证提供者

假设角色涉及使用一组可用于访问的临时安全证书AWS您通常可能无法访问的资源。这些临时凭证由访问密钥 ID、秘密访问密钥和安全令牌组成。通常，您在自己的账户中扮演角色或跨账户访问权限。

要了解有关 IAM 角色的更多信息，请参阅使用 IAM 角色在IAM 用户指南。

要了解有关担任角色的更多信息，请参阅AssumeRole在AWS安全令牌服务 API 参考。

使用以下方法配置此功能：

`credential_source`-共享AWS **`config`**文件设置

在 Amazon EC2 实例或容器中使用，用于指定 SDK 或开发工具可以在何处找到有权代入您在中指定的角色的证书`role_arn`参数。

默认值：无

有效值：

- 环境— 指定 SDK 或工具用于从环境变量中检索源凭证AWS_ACCESS_KEY_ID和AWS_SECRET_ACCESS_KEY (p. 35)。
- Ec2InstanceMetadata— 指定 SDK 或工具要使用附加到 EC2 实例配置文件的 IAM 角色以获取源证书。
- EcsContainer— 指定 SDK 或工具要使用附加到 ECS 容器的 IAM 角色以获取源证书。

不能在同一配置文件中同时指定 `credential_source` 和 `source_profile`。

在 a 中设置此项的示例`config`文件以表明证书应来自亚马逊 EC2：

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

`duration_seconds`-共享AWS **`config`**文件设置

指定角色会话的最大持续时间（以秒为单位）。

仅当配置文件指定担任角色时，此设置才适用。

默认值：3600 秒（一小时）

有效值：该值的范围可以从 900 秒（15 分钟）到为角色配置的最大会话持续时间设置（最长可以是 43200 秒或 12 小时）。有关更多信息，请参阅查看角色的最大会话持续时间设置在IAM 用户指南。

在 a 中设置此项的示例config文件:

```
duration_seconds = 43200
```

external_id-共享AWS **config**文件设置

指定第三方用于在其客户账户中代入角色的唯一标识符。

此设置仅在配置文件指定担任角色且该角色的信任策略要求值时适用ExternalId。该值映射到ExternalId传递给的参数AssumeRole配置文件指定角色时的操作。

默认值：无。

有效值：请参阅在向你授予访问权限时如何使用外部 IDAWS提供给第三方的资源在IAM 用户指南。

在 a 中设置此项的示例config文件:

```
external_id = unique_value_assigned_by_3rd_party
```

mfa_serial-共享AWS **config**文件设置

指定用户在担任角色时必须使用的多因素身份验证 (MFA) 设备的标识号或序列号。

在担任角色时，如果该角色的信任策略包含需要 MFA 身份验证的条件，则为必填项。

默认值：无。

有效值：该值可以是硬件设备的序列号（例如GAHT12345678），或者虚拟 MFA 设备的亚马逊资源名称 (ARN)。有关 MFA 的更多信息，请参阅配置受 MFA 保护的 API 访问权限在IAM 用户指南。

在 a 中设置此项的示例config文件:

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

role_arn-共享AWS **config**文件设置

指定您要用于执行使用此配置文件请求的操作的 IAM 角色的亚马逊资源名称 (ARN)。

默认值：无。

有效值：该值必须是 IAM 角色的 ARN，格式如下：arn:aws:iam::*account-id*:role/*role-name*

此外，您还必须指定一以下设置中：

- source_profile— 识别另一个配置文件，用于查找有权在此配置文件中担任该角色的凭据。
- credential_source— 使用由当前环境变量标识的凭证或附加到 Amazon EC2 实例配置文件或 Amazon ECS 容器实例的证书。

在 a 中设置此项的示例config文件:

```
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
```

role_session_name-共享AWS **config**文件设置

指定要附加到角色会话的名称。此名称显示在与此会话关联的条目的 AWS CloudTrail 日志中。

默认值：一个可选参数。如果您不提供此值，则在配置文件担任角色时会自动生成会话名称。

有效值：提供给RoleSessionName参数当AWS CLI称之为AssumeRole操作（或诸如AssumeRoleWithWebIdentity操作）代表您。该值成为您可以查询的代入角色用户 Amazon 资源名称 (ARN) 的一部分，并显示为CloudTrail此配置文件调用的操作的日志条目。

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

在 a 中设置此项的示例config文件:

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

source_profile-共享AWS **config**文件设置

指定另一个配置文件，其凭据用于代入所指定的角色role_arn在原始配置文件中设置。要了解共享中的配置文件是如何使用的AWS config和credentials文件，请参阅已共享config和credentials文件 (p. 2)。

如果您指定的配置文件也是代入角色配置文件，则将按顺序担任每个角色以完全解析凭证。当 SDK 遇到带有静态凭据的配置文件时，该链就会停止。角色链限制了你的AWS CLI要么AWSAPI 角色会话最长为一小时，且无法延长。有关更多信息，请参阅角色术语和概念在IAM 用户指南。

默认值：无。

有效值：由中定义的配置文件的名称组成的文本字符串config和credentials文件。您还必须为指定一个值role_arn在当前的个人资料中。

Note

此设置可以替代credential_source。你不能同时指定两者source_profile和credential_source在同一个个人资料中。

在配置文件中设置此项的示例：

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

[profile B]
aws_access_key_id=AKIAIOSFODNN7EXAMPLE
aws_secret_access_key=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
```

web_identity_token_file-共享AWS **config**文件设置

指定包含来自访问令牌的文件的路径支持的 OAuth 2.0 提供商要么OpenID Connect ID 身份提供商。

此设置允许使用 Web 联合身份验证提供商进行身份验证，例如谷歌，Facebook，以及亚马逊，除其他外。SDK 或开发者工具加载此文件的内容并将其作为WebIdentityToken当它调用AssumeRoleWithWebIdentity代表您进行操作。

默认值：无。

有效值：此值必须是路径和文件名。该文件必须包含身份提供商向您提供的 OAuth 2.0 访问令牌或 OpenID Connect 令牌。

在 a 中设置此项的示例config文件:

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

兼容AWSSDKS

以下 SDK 支持本页上描述的功能和设置，但请注意部分例外情况：

SDK	支持	备注或更多信息
AWS CLI v2	是	
C++ 软件开发工具包	部分	credential_source不支持。duration_seconds不支持。mfa_serial不支持。
适用于 Go V2 的 SDK (1.x)	是	
适用于 Go 1.x 的 SDK (V1)	是	
适用于 Java 的 SDK 2.x	部分	mfa_serial不支持。
适用于 Java 的 SDK 1.x	部分	mfa_serial不支持。
适用于JavaScript3.x	是	
适用于JavaScript2.x	部分	credential_source不支持。
适用于.NET 3.x 的开发工具包	是	
适用于 PHP 的 SDK 3.x	是	
适用于 Python 的 SDK (Boto3)	是	
适用于 Ruby 3.x 的 SDK	是	

容器凭证提供商

容器凭证提供者获取客户容器化应用程序的凭证。该证书提供商对亚马逊弹性容器服务 (Amazon ECS) 客户非常有用。软件开发工具包将尝试通过 GET 请求从指定的 HTTP 端点加载证书。要了解有关... 的更多信息AWS_CONTAINER_CREDENTIALS_RELATIVE_URI环境变量，请参见任务的 IAM 角色在亚马逊弹性容器服务开发者指南。

使用以下方法配置此功能：

AWS_CONTAINER_CREDENTIALS_FULL_URI-环境变量

包含完整的 HTTP 网址端点，供开发工具包在请求凭证时使用。这包括方案和主机。

默认值：无。

有效值：有效的 URI。

注意：此设置是替代设置*AWS_CONTAINER_CREDENTIALS_RELATIVE_URI*并且只有在以下情况下才会使用*AWS_CONTAINER_CREDENTIALS_RELATIVE_URI*未设置。

Linux/macOS 通过命令行设置环境变量的示例：

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

or

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

AWS_CONTAINER_CREDENTIALS_RELATIVE_URI-环境变量

指定 SDK 在请求凭证时要使用的相对 HTTP 网址端点。

默认值：无。

有效值：有效的相对 URI。

Linux/macOS 通过命令行设置环境变量的示例：

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

AWS_CONTAINER_AUTHORIZATION_TOKEN-环境变量

如果设置了此变量，SDK 将使用环境变量的值在 HTTP 请求上设置授权标头。

默认值：无。

有效值：字符串。

Linux/macOS 通过命令行设置环境变量的示例：

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

兼容AWSSDKS

以下 SDK 支持本页上描述的功能和设置，但请注意部分例外情况：

SDK	支持	备注或更多信息
AWS CLI v2	是	
C++ 软件开发工具包	是	
适用于 Go V2 的 SDK (1.x)	是	
适用于 Go 1.x 的 SDK (V1)	是	
适用于 Java 的 SDK 2.x	是	
适用于 Java 的 SDK 1.x	是	
适用于JavaScript3.x	是	

SDK	支持	备注或更多信息
适用于JavaScript2.x	是	
适用于.NET 3.x 的开发工具包	是	
适用于 PHP 的 SDK 3.x	是	
适用于 Python 的 SDK (Boto3)	是	
适用于 Ruby 3.x 的 SDK	是	

IMDS 凭证提供商

实例元数据服务 (IMDS) 提供有关您的实例的数据，您可以使用这些数据来配置或管理正在运行的实例。有关可用数据的更多信息，请参阅实例元数据和用户数据在适用于 Linux 实例的亚马逊 EC2 用户指南要么实例元数据和用户数据在适用于 Windows 实例的亚马逊 EC2 用户指南。Amazon EC2 提供了一个可供实例使用的本地终端节点，该终端节点可以为实例提供各种信息。如果实例附加了角色，则它可以提供一组对该角色有效的证书。软件开发工具包可以使用该端点来解析作为其一部分的证书默认凭证提供者链 (p. 23)。默认使用实例元数据服务版本 2 (imdsv2)，这是使用会话令牌的更安全的 IMDS 版本。如果由于不可重试的情况（HTTP 错误代码 403、404、405）而失败，则使用 imdsv1 作为后备。

使用以下方法配置此功能：

AWS_EC2_METADATA_DISABLED-环境变量

是否尝试使用亚马逊 EC2 实例元数据服务 (IMDS) 来获取证书。

默认值：`false`。

有效值：`true`，`false`。

ec2_metadata_service_endpoint-共享AWS **config**文件设置，
AWS_EC2_METADATA_SERVICE_ENDPOINT-环境变量

IMDS 的终端节点。

默认值：如果`ec2_metadata_service_endpoint_mode`等于`IPv4`，则默认终端节点为`http://169.254.169.254`。如果`ec2_metadata_service_endpoint_mode`等于`IPv6`，则默认终端节点为`http://[fd00:ec2::254]`。

有效值：有效的 URI。

ec2_metadata_service_endpoint_mode-共享AWS **config**文件设置，
AWS_EC2_METADATA_SERVICE_ENDPOINT_MODE-环境变量

IMDS 的终端节点模式。

默认值：`IPv4`。

有效值：`IPv4`，`IPv6`。

Note

IMDS 凭证提供者是其中的一部分凭证提供者链 (p. 23)。但是，只有在本系列中的其他几个提供商之后，才会对IMDS凭证提供者进行检查。因此，如果您希望您的程序使用此提供商的证书，则必须从配置中删除其他有效的凭证提供者或使用其他配置文件。或者，与其依赖凭证提供者链自动发现哪个提供商返回了有效凭证，不如在代码中指定 IMDS 凭证提供者的使用。您可以在创建服务客户端时直接指定凭据来源。

IMDS 凭证的安全性

默认情况下，当AWS软件开发工具包未配置有效凭证，该软件开发工具包将尝试使用 Amazon EC2 实例元数据服务 (IMDS) 来检索证书AWS角色。可以通过设置来禁用此行为`AWS_EC2_METADATA_DISABLED`环境变量为`true`。这样可以防止不必要的网络活动，并增强可能被模仿 Amazon EC2 实例元数据服务的不可信网络的安全性。

> **Note**
>
> AWS无论这些设置如何，配置了有效凭证的 SDK 客户端都不会使用 IMDS 检索凭证。

禁用亚马逊 EC2 IMDS 凭证的使用

如何设置此环境变量取决于所使用的操作系统以及您是否希望更改保持不变。

Linux 和 macOS

使用 Linux 或 macOS 的客户可以使用以下命令设置此环境变量：

```
$ export AWS_EC2_METADATA_DISABLED=true
```

如果您希望此设置在多个 shell 会话和系统重启中保持不变，则可以将上述命令添加到您的 shell 配置文件中，例如`.bash_profile`，`.zsh_profile`，或`.profile`。

Windows

使用 Windows 的客户可以使用以下命令设置此环境变量：

```
$ set AWS_EC2_METADATA_DISABLED=true
```

如果您希望此设置在多个 shell 会话和系统重启中保持不变，则可以改用以下命令：

```
$ setx AWS_EC2_METADATA_DISABLED=true
```

> **Note**
>
> 的setx命令不会将该值应用于当前 shell 会话，因此您需要重新加载或重新打开 shell 才能使更改生效。

与之兼容AWSSDKS

以下 SDK 支持本页上描述的功能和设置，但请注意部分例外情况：

SDK	支持	备注或更多信息
AWS CLI v2	是	
C++ 软件开发工具包	是	
适用于 Go V2 的 SDK (1.x)	是	
适用于 Go 1.x 的 SDK (V1)	是	
适用于 Java 的 SDK 2.x	是	
适用于 Java 的 SDK 1.x	是	

SDK	支持	备注或更多信息
适用于JavaScript3.x	是	
适用于JavaScript2.x	是	
适用于.NET 3.x 的开发工具包	是	
适用于 PHP 的 SDK 3.x	是	
适用于 Python 的 SDK (Boto3)	是	
适用于 Ruby 3.x 的 SDK	是	

流程凭证提供商

SDK 提供了一种扩展自定义用例的凭证提供者链的方法。

> Warning
>
> 下面介绍一种从外部流程获取凭据的方法。这可能很危险，因此请谨慎行事。如果可能，应优先选择其他证书提供者。如果使用此选项，则应确保`config`使用适用于您的操作系统的安全最佳实践，尽可能锁定文件。确认您的自定义凭证工具未向写入任何机密信息`StdErr`，因为 SDK 和AWS CLI可以捕获和记录此类信息，从而有可能将其暴露给未经授权的用户。

使用以下方法配置此功能：

`credential_process`-共享AWS **`config`**文件设置

指定 SDK 或工具代表您运行的外部命令，以生成或检索要使用的身份验证凭据。该设置指定 SDK 将调用的程序/命令的名称。当 SDK 调用该进程时，它会等待进程将 JSON 数据写入`stdout`。自定义提供程序必须以特定格式返回信息。该信息包含 SDK 或工具可用于对您进行身份验证的凭据。

> Note
>
> 流程凭证提供者是其中的一部分凭证提供者链 (p. 23)。但是，只有在本系列中的其他几个提供商之后，才会对流程凭证提供者进行检查。因此，如果您希望您的程序使用此提供商的证书，则必须从配置中删除其他有效的凭证提供者或使用其他配置文件。或者，与其依赖凭证提供者链自动发现哪个提供者返回了有效凭证，不如在代码中指定流程凭证提供者的使用。您可以在创建服务客户端时直接指定凭据来源。

指定凭证程序的路径

该设置的值是一个字符串，其中包含指向 SDK 或开发工具代表您运行的程序的路径：

- 路径和文件名只能包含以下字符：A-Z、a-z、0-9、连字符 (-)、下划线 (_)、句点 (.)、正斜杠 (/)、反斜杠 (\) 和空格。
- 如果路径或文件名包含空格，请将完整路径和文件名用双引号 (" ") 括起来。
- 如果参数名称或参数值包含空格，则用双引号 (" ") 将该元素括起来。仅括起来名称或值，而不是名称值对。
- 不要在字符串中包含任何环境变量。例如，不要包含`$HOME`要么`%USERPROFILE%`。
- 不要将主文件夹指定为~。* 必须指定完整路径或基本文件名。如果有基本文件名，则系统会尝试在指定的文件夹中查找该程序`PATH`环境变量。

以下示例显示了在共享环境中设置凭证流程`config`在 Linux/macOS 上存档。

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

以下示例显示了在共享环境中设置凭证流程configWindows 上的文件。

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

证书程序的有效输出

SDK 运行配置文件中指定的命令，然后从标准输出流中读取数据。无论是脚本还是二进制程序，您指定的命令都必须生成 JSON 输出STDOUT与以下语法相匹配。

```
{
    "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

截至撰写本文之时，Version 密钥必须设置为 1。随时间推移和该结构的发展，该值可能会增加。

的Expiration密钥是一个 RFC3339 格式的时间戳。如果Expiration该工具的输出中不存在密钥，SDK 假设凭证是不会刷新的长期凭证。否则，这些证书将被视为临时证书，它们会通过重新运行来自动刷新credential_process在证书到期之前执行命令。

Note

SDK 确实如此不像假设角色凭据一样缓存外部进程凭证。如果需要缓存，则必须在外部进程中实现。

外部进程可以返回非零返回代码，以指示在检索凭证时发生错误。

兼容AWSSDKS

以下 SDK 支持本页上描述的功能和设置，但请注意部分例外情况：

SDK	支持	备注或更多信息
AWS CLI v2	是	
C++ 软件开发工具包	是	
适用于 Go V2 的 SDK (1.x)	是	
适用于 Go 1.x 的 SDK (V1)	是	
适用于 Java 的 SDK 2.x	是	
适用于 Java 的 SDK 1.x	是	
适用于JavaScript3.x	是	

SDK	支持	备注或更多信息
适用于JavaScript2.x	是	
适用于.NET 3.x 的开发工具包	是	
适用于 PHP 3.x 的 SDK	是	
适用于 Python 的 SDK (Boto3)	是	
适用于 Ruby 3.x 的 SDK	是	

IAM 身份中心凭证提供商

此身份验证机制AWS IAM Identity Center用于获取您的代码的单点登录 (SSO) 访问AWS 服务权限。

> Note
>
> 在 AWS SDK API 文档中，IAM 身份中心凭证提供商被称为 SSO 凭证提供商。

启用 IAM Identity Center 后，您可以在共享AWS`config`文件中为其设置定义配置文件。此配置文件用于连接到 IAM 身份中心访问门户。当用户成功通过 IAM Identity Center 进行身份验证后，门户网站会返回与该用户关联的 IAM 角色的短期证书。要了解 SDK 如何从配置中获取临时证书并将其用于AWS 服务请求，请参阅了解 IAM 身份中心身份验证 (p. 10)。

通过`config`文件配置 IAM 身份中心有两种方法：

- SSO 令牌提供商配置（推荐）-将您的会话期限延长至最多七天。
- 传统不可刷新的配置-获得固定的八小时会话。

在这两种配置中，您都需要在会话到期后重新登录。

要设置自定义会话持续时间，必须使用 SSO 令牌提供程序配置。有关会话持续时间的更多信息，请参阅AWS IAM Identity Center用户指南中的配置AWS访问门户会话持续时间。

以下两份指南包含有关 IAM 身份中心的其他信息：

- AWS IAM Identity Center 用户指南
- AWS IAM Identity Center门户 API 参考

先决条件

您必须先启用 IAM 身份中心。有关启用 IAM Identity Center 身份验证的详细信息，请参阅AWS IAM Identity Center用户指南中的入门。

或者，请按照本指南中的IAM Identity Center 身份验证 (p. 9)说明进行操作。这些说明提供了完整的指导，从启用 IAM Identity Center 到完成此处接下来的必要共享`config`文件配置。

SSO 令牌提供商配置

> Note
>
> 要使用为您创建此配置，请参阅中的使用`aws configure sso`向导配置您的配置文件AWS CLI。AWS CLI

当您使用 SSO 令牌提供商配置时，您的 AWS SDK 或工具会自动刷新您的会话，直到延长的会话时长（最多七天）。

config 文件的 sso-session 部分用于对获取 SSO 访问令牌的配置变量进行分组，然后可以用来获取 AWS 凭证。有关格式化config文件中各部分的更多详细信息，请参阅配置文件的格式 (p. 3)。

您定义一个 sso-session 部分并将其关联到配置文件。必须在 sso-session 部分内设置 sso_region 和 sso_start_url。通常，sso_role_name必须在profile部分中设置sso_account_id和，这样 SDK 才能请求AWS凭证。

> **Note**
>
> 要深入了解 SDK 和工具如何使用此配置使用和刷新凭据，请参阅了解 IAM 身份中心身份验证 (p. 10)。

以下示例将软件开发工具包配置为请求 IAM 身份中心证书。它还支持自动令牌刷新。

```
[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
```

您可以跨多个sso-session配置文件重复使用配置。

```
[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
```

sso_account_id而且sso_role_name并不是所有 SSO 令牌配置场景都必需的。如果您的应用程序仅使用支持持AWS 服务有者身份验证的凭证，则不需要传统AWS凭证。持有者身份验证是一种 HTTP 身份验证方案，它使用称为持有者令牌的安全令牌。在这种情况下，不需要 sso_account_id 和 sso_role_name。请参阅个人指南，AWS 服务以确定它是否支持不记名令牌授权。

注册范围配置为的一部分sso-session。作用域是一种限制应用程序访问用户帐户的机制。OAuth 2.0应用程序可以请求一个或多个范围，并且向该应用程序发放的访问令牌仅限于授予的范围。这些范围定义了为已注册的 OIDC 客户端请求授权的权限和客户端检索的访问令牌。有关支持的访问范围选项，请参阅《AWS IAM Identity Center用户指南》中的访问范围。以下示例集sso_registration_scopes为列出账户和角色提供访问权限。

```
[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
```

身份验证令牌缓存到磁盘~/.aws/sso/cache目录下，文件名基于会话名称。

遗留的不可刷新配置

使用遗留的不可刷新配置不支持自动令牌刷新。我们建议SSO 令牌提供商配置 (p. 32)改用。

要使用旧版不可刷新的配置，您必须在配置文件中指定以下设置：

- sso_start_url
- sso_region
- sso_account_id
- sso_role_name

您可以使用sso_start_url和sso_region设置为配置文件指定用户门户。您可以使用sso_account_id和sso_role_name设置来指定权限。

以下示例在config文件中设置了四个必填值。

```
[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
```

身份验证令牌缓存到磁盘上的~/.aws/sso/cache目录下，文件名基于sso_start_url。

IAM 身份中心凭证提供商设置

使用以下方法配置此功能：

sso_start_url-共享AWS**config**文件设置

指向贵组织的 IAM 身份中心访问门户的 URL。有关 IAM Identity Center 访问门户的更多信息，请参阅AWS IAM Identity Center用户指南中的使用AWS访问门户。

要找到此值，请打开 IAM Identity Center 控制台，查看控制面板，然后找到AWS访问门户 URL。

sso_region-共享AWS**config**文件设置

其中AWS 区域包含您的 IAM 身份中心门户主机；也就是您在启用 IAM 身份中心之前选择的区域。这与您的默认AWS区域无关，也可能有所不同。

有关AWS 区域及其代码的完整列表，请参阅中的区域终端节点Amazon Web Services 一般参考。要查找此值，请打开 IAM Identity Center 控制台，查看控制面板，然后找到区域。

sso_account_id-共享AWS**config**文件设置

通过AWS Organizations服务添加AWS 账户的用于身份验证的数字 ID。

要查看可用账户列表，请前往 IAM Identity Center 控制台并打开AWS 账户页面。您还可以在AWS IAM Identity Center门户 API 参考中使用 ListAccountsAPI 方法查看可用账户列表。例如，您可以调用 “列表账户” AWS CLI 方法。

sso_role_name-共享AWS**config**文件设置

作为 IAM 角色配置的权限集的名称，用于定义用户生成的权限。该角色必须存在于AWS 账户指定的中sso_account_id。使用角色名称，而不是角色亚马逊资源名称 (ARN)。

权限集附有 IAM 策略和自定义权限策略，并定义了用户对其分配的访问权限级别AWS 账户。

要查看每个可用权限集的列表AWS 账户，请转到 IAM Identity Center 控制台并打开AWS 账户页面。选择AWS 账户表格中列出的正确权限集名称。您还可以使用 Port AWS IAM Identity Centeral API 参考中的 ListAccountRolesAPI 方法查看可用权限集列表。例如，您可以调用AWS CLI方法list-account-roles。

`sso_registration_scopes`-共享AWS**`config`**文件设置

要授权的有效范围字符串的逗号分隔列表。`sso-session`范围授权对 IAM Identity Center 持有者令牌授权终端节点的访问。要从 IAM Identity Center 服务中取回刷新令牌，`sso:account:access`必须授予最小范围。有关支持的访问范围字符串，请参阅AWS IAM Identity Center用户指南中的访问范围。此设置不适用于旧版不可刷新的配置。使用旧版配置发行的令牌被`sso:account:access`隐式限制在作用域范围内。

与 AWS SDK 的兼容性

以下 SDK 支持本页上描述的功能和设置，但请注意部分例外情况：

SDK	支持	备注或更多信息
AWS CLI v2	是	
适用于 C++ 的 SDK	是	
适用于 Go V2 的 SDK (1.x)	是	
适用于 Go 1.x 的 SDK (V1)	是	
适用于 Java 的 SDK 2.x	是	`credentials`文件中也支持配置值。
适用于 Java 的 SDK 1.x	否	
适用于 JavaScript 3.x 的软件开发工具包	是	
适用于 JavaScript 2.x 的 SDK	是	
适用于.NET 3.x 的 SDK	是	
适用于 PHP 的 SDK 3.x	是	
适用于 Python 的 SDK (Boto3)	是	
适用于 Ruby 3.x 的 SDK	是	

AWS 访问密钥

Warning

为了避免安全风险，在开发专用软件或处理真实数据时，请勿使用 IAM 用户进行身份验证。而是使用与身份提供商的联合身份验证，例如 AWS IAM Identity Center。

AWSIAM 用户的访问密钥可用作您的AWS证书。的AWSSDK 会自动使用这些AWS用于签署 API 请求的凭证AWS，这样您的工作负载就可以访问您的AWS安全便捷地提供资源和数据。建议始终使用`aws_session_token`因此证书是临时的，过期后不再有效。不建议使用长期凭证。

Note

如果AWS无法刷新这些临时证书，AWS可以延长凭证的有效期，这样您的工作负载就不会受到影响。

共享的AWS credentials文件是存储凭据信息的推荐位置，因为它安全地位于应用程序源目录之外，并且与共享的 SDK 特定设置是分开的config文件。

要了解更多相关信息AWS凭证和使用访问密钥，请参阅AWS安全凭证和管理 IAM 用户的访问密钥在IAM 用户指南。

使用以下方法配置此功能：

aws_access_key_id-共享AWS **config**文件设置, **aws_access_key_id**-共享AWS **credentials**文件设置（推荐方法）, **AWS_ACCESS_KEY_ID**-环境变量

指定AWS访问密钥用作证书的一部分，用于对用户进行身份验证。

aws_secret_access_key-共享AWS **config**文件设置, **aws_secret_access_key**-共享AWS **credentials**文件设置（推荐方法）, **AWS_SECRET_ACCESS_KEY**-环境变量

指定AWS作为证书一部分用于对用户进行身份验证的密钥。

aws_session_token-共享AWS **config**文件设置, **aws_session_token**-共享AWS **credentials**文件设置（推荐方法）, **AWS_SESSION_TOKEN**-环境变量

指定一个AWS会话令牌用作证书的一部分，用于对用户进行身份验证。您会收到此值作为成功申请代入角色所返回的临时证书的一部分。只有在手动指定临时安全凭证时才需要会话令牌。但是，我们建议您始终使用临时安全证书，而不是长期证书。有关安全建议，请参阅IAM 中的安全最佳实践。

有关如何获取这些值的说明，请参阅使用短期凭证进行身份验证 (p. 14)。

在中设置这些必需值的示例config要么credentials文件:

```
[default]
aws_access_key_id = AKIAIOSFODNN7EXAMPLE
aws_secret_access_key = wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
aws_session_token = AQoEXAMPLEH4aoAH0gNCAPy...truncated...zrkuWJOgQs8IZZaIv2BXIa2R4Olgk
```

Linux/macOS 通过命令行设置环境变量的示例：

```
export AWS_ACCESS_KEY_ID=AKIAIOSFODNN7EXAMPLE
export AWS_SECRET_ACCESS_KEY=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
export
 AWS_SESSION_TOKEN=AQoEXAMPLEH4aoAH0gNCAPy...truncated...zrkuWJOgQs8IZZaIv2BXIa2R4Olgk
```

Windows 通过命令行设置环境变量的示例：

```
setx AWS_ACCESS_KEY_ID AKIAIOSFODNN7EXAMPLE
setx AWS_SECRET_ACCESS_KEY wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
setx AWS_SESSION_TOKEN AQoEXAMPLEH4aoAH0gNCAPy...truncated...zrkuWJOgQs8IZZaIv2BXIa2R4Olgk
```

兼容AWSSDKS

以下 SDK 支持本页上描述的功能和设置，但请注意部分例外情况：

SDK	支持	备注或更多信息
AWS CLI v2	是	
C++ 软件开发工具包	是	共享的config不支持文件。

SDK	支持	备注或更多信息
适用于 Go V2 的 SDK (1.x)	是	
适用于 Go 1.x 的 SDK (V1)	是	
适用于 Java 的 SDK 2.x	是	
适用于 Java 的 SDK 1.x	是	
适用于JavaScript3.x	是	
适用于JavaScript2.x	是	
适用于.NET 3.x 的开发工具包	是	不支持环境变量。
适用于 PHP 的 SDK 3.x	是	
适用于 Python 的 SDK (Boto3)	是	
适用于 Ruby 3.x 的 SDK	是	

标准化功能

许多功能已标准化为一致的默认值，并且在许多 SDK 中以相同的方式工作。当跨多个 SDK 进行编码时，这种一致性可以提高工作效率和清晰度。所有设置都可以在代码中被覆盖，详情请参阅您的特定 SDK API。

> **Important**
>
> 并非所有 SDK 都支持所有功能，甚至不是功能中的所有方面。

主题

Amazon EC2 实例元数据

Amazon EC2 在实例上提供了一项名为实例元数据服务 (IMDS) 的服务。要了解有关此服务的更多信息，请参阅实例元数据和用户数据在适用于 Linux 实例的亚马逊 EC2 用户指南要么实例元数据和用户数据在适用于 Windows 实例的亚马逊 EC2 用户指南。尝试检索配置了 IAM 角色的 Amazon EC2 实例的证书时，与实例元数据服务的连接是可以调整的。

使用以下方法配置此功能：

metadata_service_num_attempts-共享AWS **config**文件设置, **AWS_METADATA_SERVICE_NUM_ATTEMPTS**-环境变量

此设置指定尝试从实例元数据服务检索数据时放弃之前尝试的总次数。

默认值：1

有效值：大于或等于 1 的数字。

metadata_service_timeout-共享AWS **config**文件设置, **AWS_METADATA_SERVICE_TIMEOUT**-环境变量

指定尝试从实例元数据服务检索数据时超时前的秒数。

默认值：1

有效值：大于或等于 1 的数字。

在中设置这些值的示例config文件:

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

Linux/macOS 通过命令行设置环境变量的示例：

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

Windows 通过命令行设置环境变量的示例：

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

兼容AWSSDKS

以下 SDK 支持本页上描述的功能和设置，但请注意部分例外情况：

SDK	支持	备注或更多信息
AWS CLI v2	是	
C++ 软件开发工具包	否	
适用于 Go V2 的 SDK (1.x)	否	
适用于 Go 1.x 的 SDK (V1)	否	
适用于 Java 的 SDK 2.x	否	
适用于 Java 的 SDK 1.x	部分	metadata_service_num_attempts不支持。
适用于JavaScript3.x	否	
适用于JavaScript2.x	否	

SDK	支持	备注或更多信息
适用于.NET 3.x 的开发工具包	否	
适用于 PHP 的 SDK 3.x	是	
适用于 Python 的 SDK (Boto3)	是	
适用于 Ruby 3.x 的 SDK	否	

Amazon S3 接入点

Amazon S3 服务提供接入点作为与 Amazon S3 存储桶交互的替代方式。接入点具有独特的策略和配置，可以应用于接入点而不是直接应用于存储桶。与AWS软件开发工具包，您可以在存储桶字段中使用接入点亚马逊资源名称 (ARN) 进行 API 操作，而不必明确指定存储桶名称。它们用于特定操作，例如使用接入点 ARN 和`GetObject`从存储桶中获取对象，或者使用接入点 ARN 和`PutObject`向存储桶添加对象。

要了解有关 Amazon S3 接入点和 ARN 的更多信息，请参阅使用接入点在亚马逊 S3 用户指南。

使用以下方法配置此功能：

`s3_use_arn_region`-共享AWS **`config`**文件设置, **`AWS_S3_USE_ARN_REGION`**-环境变量, 要直接在代码中配置值，请直接咨询您的特定 SDK。

> 此设置控制 SDK 是否使用接入点 ARNAWS 区域为请求构建区域终端节点。软件开发工具包会验证 ARNAWS 区域由相同的方式提供AWS按照客户端的配置进行分区AWS 区域以防止最有可能失败的跨分区调用。如果定义了 multiply，则优先使用代码配置的设置，其次是环境变量设置。
>
> 默认值：`false`
>
> 有效值：
>
> - **`true`**— 软件开发工具包使用 ARNAWS 区域构造端点而不是客户端配置的端点时AWS 区域。例外：如果已配置客户端AWS 区域是 FIPSAWS 区域，那么它必须与 ARN 相匹配AWS 区域。否则将导致出现错误。
> - **`false`**— SDK 使用客户端配置的AWS 区域在构造端点时。

兼容AWSSDKS

以下 SDK 支持本页上描述的功能和设置，但请注意部分例外情况：

SDK	支持	备注或更多信息
AWS CLI v2	是	
C++ 软件开发工具包	是	
适用于 Go V2 的 SDK (1.x)	是	
适用于 Go 1.x 的 SDK (V1)	是	
适用于 Java 的 SDK 2.x	是	
适用于 Java 的 SDK 1.x	是	

SDK	支持	备注或更多信息
适用于JavaScript3.x	是	
适用于JavaScript2.x	是	
适用于.NET 3.x 的开发工具包	是	不遵循标准优先级；共享config文件值优先于环境变量。
适用于 PHP 的 SDK 3.x	部分	
适用于 Python 的 SDK (Boto3)	是	
适用于 Ruby 3.x 的 SDK	是	

Amazon S3 多区域访问点

Amazon S3 多区域接入点提供了一个全局终端节点，应用程序可以使用该终端节点来满足来自位于多个 Amazon S3 存储桶的请求AWS 区域。您可以使用多区域接入点来构建具有与单个区域相同的架构的多区域应用程序，然后在世界任何地方运行这些应用程序。

要了解有关多区域接入点的更多信息，请参阅亚马逊 S3 中的多区域接入点在亚马逊 S3 用户指南。

要了解有关多区域接入点亚马逊资源名称 (ARN) 的更多信息，请参阅使用多区域接入点发出请求在亚马逊 S3 用户指南。

要了解有关创建多区域接入点的更多信息，请参阅管理多区域接入点在亚马逊 S3 用户指南。

Sigv4a 算法是用于签署全局区域请求的签名实现。该算法是由 SDK 通过依赖来获得的AWS通用运行时 (CRT) 库 (p. 72)。

使用以下方法配置此功能：

s3_disable_multiregion_access_points-共享AWS **config**文件设置，**AWS_S3_DISABLE_MULTIREGION_ACCESS_POINTS**-环境变量，要直接在代码中配置值，请直接咨询您的特定 SDK。

此设置控制 SDK 是否可能尝试跨区域请求。如果定义了 multiply，则优先使用代码配置的设置，其次是环境变量设置。

默认值：`false`

有效值：

- **true**— 停止使用跨区域请求。
- **false**— 使用多区域接入点启用跨区域请求。

兼容AWSSDKS

以下 SDK 支持本页上描述的功能和设置，但请注意部分例外情况：

SDK	支持	备注或更多信息
AWS CLI v2	是	

SDK	支持	备注或更多信息
C++ 软件开发工具包	是	
适用于 Go V2 的 SDK (1.x)	是	
适用于 Go 1.x 的 SDK (V1)	否	
适用于 Java 的 SDK 2.x	是	
适用于 Java 的 SDK 1.x	否	
适用于JavaScript3.x	是	
适用于JavaScript2.x	否	
适用于.NET 3.x 的开发工具包	是	
适用于 PHP 的 SDK 3.x	是	
适用于 Python 的 SDK (Boto3)	是	
适用于 Ruby 3.x 的 SDK	是	

AWS 区域

AWS 区域是使用时需要理解的重要概念AWS 服务。

随着AWS 区域，你可以访问AWS 服务实际居住在特定的地理区域。这对于将您的数据和应用程序保持在您和您的用户访问它们的地方附近运行非常有用。区域提供容错能力、稳定性和弹性，还可以减少延迟。使用 Regions，您可以创建冗余资源，这些资源保持可用且不受区域中断的影响。

大多数AWS 服务请求与特定地理区域相关联。除非您明确使用某个区域提供的复制功能，否则您在一个区域创建的资源不存在于任何其他区域AWS 服务。例如，Amazon S3 和 Amazon EC2 支持跨区域复制。某些服务（例如 IAM）没有区域资源。

的AWS 一般参考包含有关以下内容的信息：

- 要了解区域和终端节点之间的关系并查看现有区域终端节点的列表，请参阅AWS服务端点。
- 查看所有支持的区域和每个区域的终端节点的当前列表AWS 服务，请参阅服务终端节点和配额。

创建服务客户端

以编程方式访问AWS 服务，SDK 对每个 SDK 使用一个客户端类/对象AWS 服务。例如，如果您的应用程序需要访问 Amazon EC2，则您的应用程序将创建一个 Amazon EC2 客户端对象来与该服务交互。

如果没有为客户端明确指定区域，则客户端将默认使用通过以下方式设置的区域`region`设置。但是，可以为任何单个客户端对象显式设置客户端的活动区域。以这种方式设置区域优先于该特定服务客户端的任何全局设置。备用区域是在该客户端的实例化过程中指定的，该区域特定于您的 SDK（请查看您的特定 SDK 指南或 SDK 的代码库）。

使用以下方法配置此功能：

`region`-共享AWS **`config`**文件设置, **`AWS_REGION`**-环境变量

指定默认值AWS 区域用于AWS请求。此区域用于未提供特定区域的 SDK 服务请求。

默认值：无。您必须明确指定此值。

有效值：

- 所选服务可用的任何地区代码，如中所列AWS服务端点在AWS一般参考。例如，该值us-east-1将终端节点设置为AWS 区域美国东部（弗吉尼亚北部）。
- aws-global为除区域终端节点之外还支持单独的全局终端节点的服务指定全局终端节点，例如AWS Security Token Service(AWS STS）和亚马逊简单存储服务（亚马逊 S3）。

在中设置此值的示例config文件:

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

Linux/macOS 通过命令行设置环境变量的示例：

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

Windows 通过命令行设置环境变量的示例：

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

大多数 SDK 都有一个 “配置” 对象，可用于在应用程序代码中设置默认区域。有关详细信息，请参阅您的具体内容AWSSDK 开发者指南。

与之兼容AWSSDKS

以下 SDK 支持本页上描述的功能和设置，但请注意部分例外情况：

SDK	支持	备注或更多信息
AWS CLI v2	是	AWS CLIV2 使用中的任何值AWS_REGION在任何值之前AWS_DEFAULT_REGION（两个变量都被选中）。
AWS CLIv1	是	此 SDK 使用名为的环境变量AWS_DEFAULT_REGION为此目的。
C++ 软件开发工具包	是	
适用于 Go V2 的 SDK (1.x)	是	
适用于 Go 1.x 的 SDK (V1)	是	
适用于 Java 的 SDK 2.x	是	
适用于 Java 的 SDK 1.x	是	
适用于JavaScript3.x	是	
适用于JavaScript2.x	是	
适用于.NET 3.x 的开发工具包	是	
适用于 PHP 的 SDK 3.x	是	
适用于 Python 的 SDK (Boto3)	是	此 SDK 使用名为的环境变量AWS_DEFAULT_REGION为此目的。
适用于 Ruby 3.x 的 SDK	是	

AWS STS区域化终端节点

默认情况下，AWS Security Token Service(AWS STS) 可作为全局服务使用，并且所有AWS STS请求都发送到的单个终端节点`https://sts.amazonaws.com`。全球请求映射到美国东部（弗吉尼亚北部）区域。AWS建议使用区域AWS STS终端节点而不是全球终端节点。有关AWS STS终端节点的更多信息，请参阅AWS Security Token Service API 参考中的终端节点。

使用以下方法配置此功能：

`sts_regional_endpoints`-共享AWS**`config`**文件设置, **`AWS_STS_REGIONAL_ENDPOINTS`**-环境变量

此设置指定 SDK 或工具如何确定用于与 AWS Security Token Service (AWS STS) 通信的AWS 服务端点。

默认值：`legacy`

Note

2022 年 7 月之后发布的所有新 SDK 主要版本都将默认为`regional`。新的 SDK 主要版本可能会删除此设置并使用`regional`行为。为了减少此变更对未来的影响，我们建议您尽可能开始`regional`在应用程序中使用。

有效值：（推荐值：*`regional`*）

- **`legacy`**— 对以下AWS区域使用全局AWS STS终端节点：`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`、和`us-west-2`。`sts.amazonaws.com`所有其他区域自动使用其各自的区域端点。
- **`regional`**— SDK 或工具始终使用当前配置区域的AWS STS终端节点。例如，如果将客户端配置为使用`us-west-2`，则对的所有调用都将AWS STS发送到区域终端节点`sts.us-west-2.amazonaws.com`，而不是全球`sts.amazonaws.com`终端节点。要在启用此设置时向全局终端节点发送请求，您可以将区域设置为 `aws-global`。

在`config`文件中设置这些值的示例：

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

Linux/macOS 通过命令行设置环境变量的示例：

```
export AWS_STS_REGIONAL_ENDPOINTS=regional
```

Windows 通过命令行设置环境变量的示例：

```
setx AWS_STS_REGIONAL_ENDPOINTS regional
```

与 AWS SDK 的兼容性

以下 SDK 支持本页上描述的功能和设置，但请注意部分例外情况：

SDK	支持	备注或更多信息
AWS CLI v2	部分	默认值为 `regional`。

SDK	支持	备注或更多信息
适用于 C++ 的 SDK	部分	不支持环境变量和config文件设置。SDK 通过regional设置执行。
适用于 Go V2 的 SDK (1.x)	是	
适用于 Go 1.x 的 SDK (V1)	是	
适用于 Java 的 SDK 2.x	是	
适用于 Java 的 SDK 1.x	是	
适用于 JavaScript 3.x 的软件开发工具包	是	
适用于 JavaScript 2.x 的 SDK	是	
适用于.NET 3.x 的 SDK	是	
适用于 PHP 的 SDK 3.x	是	
适用于 Python 的 SDK (Boto3)	是	
适用于 Ruby 3.x 的 SDK	是	

双栈和 FIPS 端点

使用以下方法配置此功能：

use_dualstack_endpoint-共享AWS **config**文件设置, **AWS_USE_DUALSTACK_ENDPOINT**-环境变量

开启或关闭 SDK 是否向双堆栈终端节点发送请求。要了解有关同时支持 IPv4 和 IPv6 流量的双栈终端节点的更多信息，请参阅使用亚马逊 S3 双堆栈终端节点在亚马逊简单存储服务用户指南。某些地区的某些服务可以使用双栈终端节点。

默认值：`false`

有效值：

- **true**— SDK 或工具将尝试使用双堆栈端点发出网络请求。如果服务不存在双栈端点和/或AWS 区域，则请求将失败。
- **false**— SDK 或工具不会使用双堆栈端点发出网络请求。

use_fips_endpoint-共享AWS **config**文件设置, **AWS_USE_FIPS_ENDPOINT**-环境变量

开启或关闭 SDK 或工具是否向符合 FIPS 标准的端点发送请求。联邦信息处理标准 (FIPS) 是美国政府对数据及其加密的一系列安全要求。政府机构、合作伙伴以及希望与联邦政府开展业务的机构必须遵守FIPS指导方针。与标准 AWS 端点不同，FIPS 端点使用符合 FIPS 140-2 的 TLS 软件库。要详细了解通过以下方式指定 FIPS 端点的其他方法AWS 区域，请参阅按服务分类的 FIPS 终端节点。有关亚马逊弹性计算云服务终端节点的更多信息，请参阅双栈（IPv4 和 IPv6）端点在亚马逊 EC2 API 参考。

默认值：`false`

有效值：

- **true**— SDK 或工具将向符合 FIPS 的终端节点发送请求。
- **false**— SDK 或工具不会向符合 FIPS 的端点发送请求。

与之兼容AWSSDKS

以下 SDK 支持本页上描述的功能和设置，但请注意部分例外情况：

SDK	支持	备注或更多信息
AWS CLI v2	是	
C++ 软件开发工具包	是	
适用于 Go V2 的 SDK (1.x)	是	
适用于 Go 1.x 的 SDK (V1)	是	
适用于 Java 的 SDK 2.x	是	
适用于 Java 的 SDK 1.x	否	
适用于JavaScript3.x	是	
适用于JavaScript2.x	是	
适用于.NET 3.x 的开发工具包	是	
适用于 PHP 的 SDK 3.x	是	
适用于 Python 的 SDK (Boto3)	是	
适用于 Ruby 3.x 的 SDK	是	

终端节点发现

SDK 使用端点发现来访问服务端点（用于访问各种资源的 URL），同时仍然保持灵活性AWS根据需要更改 URL。这样，您的代码就可以自动检测新的端点。某些服务没有固定的终端节点。相反，你可以在运行时通过请求先获取端点来获得可用的终端节点。检索可用端点后，代码随后使用该端点访问其他操作。例如，对于亚马逊 Timestream，软件开发工具包创建了`DescribeEndpoints`请求检索可用的终端节点，然后使用这些端点完成特定操作，例如`CreateDatabase`要么`CreateTable`。

在某些服务中，端点发现是必需的，而在另一些服务中则是可选的。它默认为任一值`true`要么`false`取决于服务是否需要端点发现。例如，Timestream 默认为`true`，而亚马逊 DynamoDB 默认为`false`。对于不需要端点发现的服务，则不启用端点发现。取而代之的是，配置选项可通过环境变量获得，共享的AWS `config`文件或 SDK 代码结构（例如，配置类）。对于需要发现端点的操作，SDK 会自动尝试发现终端节点。

使用以下方法配置此功能：

`endpoint_discovery_enabled`-共享AWS **`config`**文件设置, **`AWS_ENABLE_ENDPOINT_DISCOVERY`**-环境变量, 要直接在代码中配置值，请直接咨询您的特定 SDK。

为端点发现是可选的服务启用/禁用端点发现。某些服务需要端点发现。

默认值：`false`

有效值：

- **`true`**— 对于端点发现是可选的服务，SDK 应自动尝试发现终端节点。
- **`false`**— 对于端点发现是可选的服务，SDK 不应自动尝试发现终端节点。

与之兼容AWSSDKS

以下 SDK 支持本页上描述的功能和设置，但请注意部分例外情况：

SDK	支持	备注或更多信息
AWS CLI v2	是	
C++ 软件开发工具包	是	
适用于 Go V2 的 SDK (1.x)	是	
适用于 Go 1.x 的 SDK (V1)	是	
适用于 Java 的 SDK 2.x	是	
适用于 Java 的 SDK 1.x	是	
适用于JavaScript3.x	是	
适用于JavaScript2.x	是	
适用于.NET 3.x 的开发工具包	是	
适用于 PHP 的 SDK 3.x	是	
适用于 Python 的 SDK (Boto3)	是	
适用于 Ruby 3.x 的 SDK	是	

常规配置设置

SDK 支持一些用于配置 SDK 整体行为的常规设置。

使用以下方法配置此功能：

api_versions-共享AWS **config**文件设置

某些 AWS 服务维护多个 API 版本以支持向后兼容性。默认情况下，SDK 和AWS CLI操作使用最新的可用的 API 版本。要要求使用特定的 API 版本来处理您的请求，请附上api_versions在您的个人资料中设置。

默认值：无。（软件开发工具包使用最新的 API 版本。）

有效值：这是一个嵌套设置，后面有一行或多行缩进，每行标识一行AWS服务和要使用的 API 版本。有关信息，请参阅文档AWS服务以了解有哪些 API 版本可用。

该示例为两个设置了特定的 API 版本AWS中的服务config文件。这些 API 版本仅用于在包含这些设置的配置文件下运行的命令。任何其他服务的命令都使用该服务 API 的最新版本。

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

ca_bundle-共享AWS **config**文件设置, **AWS_CA_BUNDLE**-环境变量

指定自定义证书捆绑包的路径（带有.pem扩展），用于建立 SSL/TLS 连接。

默认值：无

有效值：指定完整路径或基本文件名。如果有基本文件名，则系统会尝试在指定的文件夹中查找该程序PATH环境变量。

在中设置此值的示例config文件:

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

Linux/macOS 通过命令行设置环境变量的示例：

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

Windows 通过命令行设置环境变量的示例：

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

output-共享AWS **config**文件设置

指定结果的格式化方式AWS CLI和其他AWS软件开发工具包和工具。

默认值：json

有效值：

- **json** – 输出采用 JSON 字符串的格式。
- **yaml** – 输出采用 YAML 字符串的格式。
- **yaml-stream** – 输出被流式处理并采用 YAML 字符串的格式。串流支持更快地处理大型数据类型。
- **text** – 输出采用多个制表符分隔字符串值行的格式。这对于将输出传递到文本处理器（如 grep、sed 或 awk）很有用。
- **table** – 输出采用表格形式，使用字符 +|- 以形成单元格边框。它通常以“人性化”格式呈现信息，这种格式比其他格式更容易阅读，但从编程方面来讲不是那么有用。

parameter_validation-共享AWS **config**文件设置

指定 SDK 或工具在将命令行参数发送到命令行参数之前是否尝试对其进行验证AWS服务端点。

默认值：true

有效值：

- **true** – 默认值。SDK 或工具对命令行参数执行客户端验证。这有助于 SDK 或工具确认参数是否有效，并捕获一些错误。在向 SDK 或工具发送请求之前，SDK 或工具可以拒绝无效的请求AWS服务端点。
- **false**— SDK 或工具在将命令行参数发送到之前不会对其进行验证AWS服务端点。的AWS服务端点负责验证所有请求并拒绝无效的请求。

兼容AWSSDKS

以下 SDK 支持本页上描述的功能和设置，但请注意部分例外情况：

SDK	支持	备注或更多信息
AWS CLI v2	部分	api_versions不支持。

SDK	支持	备注或更多信息
C++ 软件开发工具包	是	
适用于 Go V2 的 SDK (1.x)	部分	`api_versions`和`parameter_validation`不支持。
适用于 Go 1.x 的 SDK (V1)	部分	`api_versions`和`parameter_validation`不支持。
适用于 Java 的 SDK 2.x	否	
适用于 Java 的 SDK 1.x	否	
适用于JavaScript3.x	是	
适用于JavaScript2.x	是	
适用于.NET 3.x 的开发工具包	否	
适用于 PHP 的 SDK 3.x	是	
适用于 Python 的 SDK (Boto3)	是	
适用于 Ruby 3.x 的 SDK	是	

IMDS 客户端

软件开发工具包使用面向会话的请求实现实例元数据服务版本 2 (imdsv2) 客户端。有关 imdsv2 的更多信息，请参阅使用 imdsv2在适用于 Linux 实例的亚马逊 EC2 用户指南要么使用 imdsv2在适用于 Windows 实例的亚马逊 EC2 用户指南。IMDS 客户端可通过 SDK 代码库中提供的客户端配置对象进行配置。

使用以下方法配置此功能：

`retries`-客户端配置对象成员

任何失败请求的额外重试次数。

默认值：3

有效值：大于 0 的数字。

`port`-客户端配置对象成员

终端节点的端口。

默认值：80

有效值：数字。

`token_ttl`-客户端配置对象成员

代币的 TTL。

默认值：21,600 秒（6 小时，分配的最长时间）。

有效值：数字。

`endpoint`-客户端配置对象成员

IMDS 的终端节点。

默认值：如果endpoint_mode等于IPv4，则默认终端节点为http://169.254.169.254。如果endpoint_mode等于IPv6，则默认终端节点为http://[fd00:ec2::254]。

有效值：有效的 URI。

大多数 SDK 都支持以下选项。有关详细信息，请参阅您的特定 SDK 代码库。

endpoint_mode-客户端配置对象成员

IMDS 的终端节点模式。

默认值：IPv4

有效值：IPv4、IPv6

http_open_timeout-客户端配置对象成员（名称可能有所不同）

等待连接打开的秒数。

默认值：1 秒。

有效值：大于 0 的数字。

http_read_timeout-客户端配置对象成员（名称可能有所不同）

读取一块数据的秒数。

默认值：1 秒。

有效值：大于 0 的数字。

http_debug_output-客户端配置对象成员（名称可能有所不同）

设置用于调试的输出流。

默认值：无。

有效值：有效的 I/O 流，例如 STDOUT。

backoff-客户端配置对象成员（名称可能有所不同）

在两次重试之间休眠的秒数，或者客户提供的回退功能可供调用。这会覆盖默认的指数回退策略。

默认值：因软件开发工具包而异。

有效值：因软件开发工具包而异。可以是数值，也可以是对自定义函数的调用。

兼容AWSSDKS

以下 SDK 支持本页上描述的功能和设置，但请注意部分例外情况：

SDK	支持	备注或更多信息
AWS CLI v2	是	
C++ 软件开发工具包	否	IMDSv2 仅在内部使用。请参阅IMDS 凭证提供商 (p. 28)。
适用于 Go V2 的 SDK (1.x)	是	

SDK	支持	备注或更多信息
适用于 Go 1.x 的 SDK (V1)	是	
适用于 Java 的 SDK 2.x	是	
适用于 Java 的 SDK 1.x	是	
适用于JavaScript3.x	是	
适用于JavaScript2.x	是	
适用于.NET 3.x 的开发工具包	是	
适用于 PHP 的 SDK 3.x	是	
适用于 Python 的 SDK (Boto3)	是	
适用于 Ruby 3.x 的 SDK	是	

重试行为

重试行为包括有关软件开发工具包如何尝试从向其发出请求而导致的失败中恢复的设置。AWS 服务

使用以下方法配置此功能：

`max_attempts`-共享AWS**`config`**文件设置, **`AWS_MAX_ATTEMPTS`**-环境变量

指定对请求进行的最大尝试次数。

默认值：如果未指定此值，则其默认值取决于`retry_mode`设置的值：

- 如果`retry_mode`是 `legacy` — 使用特定于 SDK 的默认值（请查看您的特定 SDK 指南或 SDK 的代码库以了解`max_attempts`默认值）。
- 如果`retry_mode`是 `standard` — 尝试三次。
- 如果`retry_mode`是 `adaptive` — 尝试三次。

有效值：大于 0 的数字。

`retry_mode`-共享AWS**`config`**文件设置, **`AWS_RETRY_MODE`**-环境变量

指定 SDK 或开发者工具如何尝试重试。

默认值：`legacy`是默认的重试策略。

有效值：

- `legacy`— 特定于您的 SDK（请查看您的特定 SDK 指南或 SDK 的代码库）。
- `standard`— 跨AWS软件开发工具包的标准重试规则集。此模式包括一组标准的重试错误，并支持重试配额。除非`max_attempts`明确配置，否则此模式下默认的最大尝试次数为三次。
- `adaptive`— 一种实验性重试模式，包括标准模式的功能，但包括自动客户端限制。由于此模式是实验性的，因此将来可能会改变行为。

以下是两种模式`standard`和`adaptive`重试模式的高级伪代码：

```
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)
  }
}
```

以下是有关伪代码中使用的组件的更多详细信息：

`GetSendToken`:

令牌桶仅在`adaptive`重试模式下使用。令牌存储桶要求令牌可用才能启动请求，从而强制执行最高请求速率。SDK 客户端可配置为快速失败请求或在令牌可用之前进行阻止。

客户端速率限制是一种算法，它最初允许以不超过令牌限额的任何速率发出请求。但是，在检测到受限的响应后，客户端就会 rate-of-request 受到相应的限制。如果收到成功的回复，代币限额也会相应增加。

通过自适应速率限制，SDK 可以降低请求的发送速率，以便更好地容纳的AWS 服务容量。

`SendHTTPRequest`:

大多数 AWS SDK 使用使用连接池的 HTTP 库，这样您就可以在发出 HTTP 请求时重复使用现有连接。通常，由于限制错误，在重试请求时会重复使用连接。由于暂时性错误，在重试时不会重复使用请求。

`RequestBookkeeping`:

如果请求成功，则应更新重试配额。仅在`adaptive`重试模式下，状态变`maxsendrate`量会根据收到的响应类型进行更新。

`Retryable`:

此步骤根据以下内容确定是否可以重试响应：

- HTTP 状态代码 。
- 从服务返回的错误代码。
- 连接错误，定义为 SDK 收到的任何错误，其中未收到来自服务的 HTTP 响应。

临时错误（HTTP 状态代码 400、408、500、502、503 和 504）和限制错误（HTTP 状态代码 400、403、429、502、503 和 509）都可能被重试。SDK 重试行为是结合错误代码或服务中的其他数据确定的。

`MAX_ATTEMPTS`:

由`config`文件设置或环境变量指定。

`HasRetryQuota`

此步骤要求在重试配额存储桶中提供令牌，从而限制重试请求。重试配额存储桶是一种防止不太可能成功的重试的机制。这些配额依赖于 SDK，通常依赖于客户端，有时甚至依赖于服务端点。当请求由于各种原因失败时，可用的重试配额令牌将被移除，并在请求成功时补充。当没有剩余令牌时，将退出重试循环。

`ExponentialBackoff`

对于可以重试的错误，使用截断的指数退避来计算重试延迟。SDK 使用带抖动的截断二进制指数回退。以下算法显示了如何为请求的响应定义睡眠时间（以秒为单位）i：

```
seconds_to_sleep_i = min(b*r^i, MAX_BACKOFF)
```

在上述算法中，以下值适用：

`b = random number within the range of: 0 <= b <= 1`

`r = 2`

`MAX_BACKOFF = 20 seconds`对于大多数 SDK。请参阅您的特定 SDK 指南或源代码进行确认。

与 AWS SDK 的兼容性

以下 SDK 支持本页上描述的功能和设置，但请注意部分例外情况：

SDK	支持	备注或更多信息
AWS CLI v2	是	
适用于 C++ 的 SDK	是	
适用于 Go V2 的 SDK (1.x)	是	
适用于 Go 1.x 的 SDK (V1)	否	
适用于 Java 的 SDK 2.x	是	
适用于 Java 的 SDK 1.x	是	
适用于 JavaScript 3.x 的软件开发工具包	是	
适用于 JavaScript 2.x 的 SDK	否	支持最大重试次数、带抖动的指数退避以及用于重试退避的自定义方法选项。
适用于.NET 3.x 的 SDK	是	
适用于 PHP 的 SDK 3.x	是	
适用于 Python 的 SDK (Boto3)	是	
适用于 Ruby 3.x 的 SDK	是	

特定于服务的终端节点

服务特定的终端节点配置提供了使用您选择的终端节点来处理 API 请求并保留该选择的选项。这些设置为支持本地终端节点、VPC 终端节点和第三方本地端点提供了灵活性AWS开发环境。不同的端点可用于测试和生产环境。您可以为个人指定终端节点 URLAWS 服务。

使用以下方法配置此功能：

`endpoint_url`-共享AWS **`config`**文件设置, **`AWS_ENDPOINT_URL`**-环境变量

> 当直接在配置文件中指定或作为环境变量指定时，此设置将指定用于所有服务请求的端点。此端点会被任何已配置的服务特定端点覆盖。

您也可以在中使用此设置services共享的部分AWS config文件，用于为特定服务设置自定义终端节点。有关用于子部分的所有服务标识符密钥的列表services部分，请参阅服务特定端点的标识符 (p. 56)。

默认值：none

有效值：包含终端节点的方案和主机的 URL。URL 可以选择包含包含一个或多个路径段的路径组件。

AWS_ENDPOINT_URL_<SERVICE>-环境变量

AWS_ENDPOINT_URL_<SERVICE>，哪里<SERVICE>是AWS 服务标识符，为特定服务设置自定义终端节点。有关所有特定于服务的环境变量的列表，请参见服务特定端点的标识符 (p. 56)。

此特定于服务的端点会覆盖中设置的任何全局终端节点AWS_ENDPOINT_URL。

默认值：none

有效值：包含终端节点的方案和主机的 URL。URL 可以选择包含包含一个或多个路径段的路径组件。

ignore_configured_endpoint_urls-共享AWS **config**文件设置，
AWS_IGNORE_CONFIGURED_ENDPOINT_URLS-环境变量

此设置用于忽略所有自定义终端节点配置。

请注意，无论此设置如何，都将使用代码中或服务客户端本身上设置的任何显式端点。例如，包括--endpoint-url带有的命令行参数AWS CLI命令或将端点 URL 传递给客户端构造函数将始终生效。

默认值：false

有效值：

- **true**— SDK 或工具不会从共享版本中读取任何自定义配置选项config文件或来自环境变量，用于设置端点 URL。
- **false**— SDK 或工具使用共享端点中用户提供的任何可用端点config文件或来自环境变量。

使用环境变量配置端点

要将所有服务的请求路由到自定义终端节点 URL，请设置AWS_ENDPOINT_URL全局环境变量。

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

路由特定请求的路由AWS 服务要访问自定义终端节点 URL，请使用AWS_ENDPOINT_URL_<SERVICE>环境变量。Amazon DynamoDB有一个serviceId的DynamoDB。对于此服务，终端节点 URL 环境变量为AWS_ENDPOINT_URL_DYNAMODB。此端点优先于中设置的全局端点AWS_ENDPOINT_URL对于这项服务。

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

再举一个例子，AWS Elastic Beanstalk有一个serviceId的Elastic Beanstalk。的AWS 服务标识符基于 API 模型的serviceId用下划线替换所有空格并将所有字母大写。要设置此服务的终端节点，相应的环境变量是AWS_ENDPOINT_URL_ELASTIC_BEANSTALK。有关所有特定于服务的环境变量的列表，请参见服务特定端点的标识符 (p. 56)。

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

使用共享配置端点config文件

在共享中config文件，endpoint_url在不同的地方用于不同的功能。

- endpoint_url直接在 a 中指定profile使该端点成为全局端点。
- endpoint_url嵌套在 a 中的服务标识符密钥下services部分使该端点仅适用于向该服务发出的请求。有关定义的详细信息services您共享的分区config文件，请参阅配置文件的格式 (p. 3)。

以下示例使用了services定义配置用于 Amazon S3 的服务专用终端节点 URL 和用于所有其他服务的自定义全局终端节点：

```
[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
```

单个配置文件可以为多个服务配置终端节点。此示例说明如何为 Amazon S3 设置服务特定的终端节点 URL 以及AWS Elastic Beanstalk在同一个个人资料中。AWS Elastic Beanstalk有一个serviceId的Elastic Beanstalk。的AWS 服务标识符基于 API 模型的serviceId用下划线替换所有空格并将所有字母小写。因此，服务标识符密钥变为elastic_beanstalk并且此服务的设置已上线elastic_beanstalk = 。有关要在中使用的所有服务标识符密钥的列表services部分，请参阅服务特定端点的标识符 (p. 56)。

```
[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
```

服务配置部分可以从多个配置文件中使用。例如，两个配置文件可以使用相同的配置文件services在更改其他配置文件属性时定义：

```
[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
```

使用基于角色的凭据在配置文件中配置端点

如果您的个人资料具有通过配置的基于角色的凭证source_profile参数用于 IAM 假设角色功能，SDK 仅使用指定配置文件的服务配置。它不使用与其关联的角色配置文件。例如，使用以下共享config文件:

```
[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
```

如果你使用个人资料B然后在你的代码中调用 Amazon EC2，终端节点解析为`https://profile-b-ec2-endpoint.aws`。如果您的代码向任何其他服务发出请求，则终端节点解析将不遵循任何自定义逻辑。端点无法解析到配置文件中定义的全局终端节点A。让全局端点对配置文件生效B，你需要设置`endpoint_url`直接在个人资料中B。有关 `source_profile` 设置的更多信息，请参阅扮演角色凭证提供者 (p. 23)。

设置的优先级

此功能的设置可以同时使用，但每项服务只有一个值会优先使用。对于对给定的 API 调用AWS 服务，则使用以下顺序来选择值：

1. 在代码中或服务客户端本身上设置的任何显式设置都优先于其他任何设置。
   - 对于AWS CLI，这是提供的值`--endpoint-url`命令行参数。对于 SDK，显式分配可以采用您在实例化 SDK 时设置的参数的形式AWS 服务客户端或配置对象。
2. 由特定于服务的环境变量提供的值，例如`AWS_ENDPOINT_URL_DYNAMODB`。
3. 提供的值`AWS_ENDPOINT_URL`全局端点环境变量。
4. 提供的值`endpoint_url`设置嵌套在中的服务标识符密钥下`services`共享的部分`config`文件。
5. 提供的值`endpoint_url`直接在 a 中指定设置`profile`的共享`config`文件。
6. 相应端点的任何默认终端节点 URLAWS 服务是最后使用的。

与之兼容AWSSDKS

以下 SDK 支持本页上描述的功能和设置，但请注意部分例外情况：

SDK	支持	备注或更多信息
AWS CLI v2	是	
C++ 软件开发工具包	否	
适用于 Go V2 的 SDK (1.x)	否	
适用于 Go 1.x 的 SDK (V1)	否	
适用于 Java 的 SDK 2.x	否	
适用于 Java 的 SDK 1.x	否	
适用于JavaScript3.x	否	
适用于JavaScript2.x	否	
适用于.NET 3.x 的开发工具包	是	
适用于 PHP 的 SDK 3.x	否	
适用于 Python 的 SDK (Boto3)	是	
适用于 Ruby 3.x 的 SDK	是	

服务特定端点的标识符

有关如何以及在何处使用下表中的标识符的信息，请参阅特定于服务的终端节点 (p. 52)。

serviceId	共享的服务标识密钥 AWS **config** 文件	**AWS_ENDPOINT_URL_<SERVICE>** 环境变量
AccessAnalyzer	accessanalyzer	AWS_ENDPOINT_URL_ACCESSANALYZER
Account	account	AWS_ENDPOINT_URL_ACCOUNT
ACM	acm	AWS_ENDPOINT_URL_ACM
ACM PCA	acm_pca	AWS_ENDPOINT_URL_ACM_PCA
Alexa For Business	alexaforbusiness	AWS_ENDPOINT_URL_ALEXA_FOR_BUSINESS
amp	amp	AWS_ENDPOINT_URL_AMP
Amplify	amplify	AWS_ENDPOINT_URL_AMPLIFY
AmplifyBackend	amplifybackend	AWS_ENDPOINT_URL_AMPLIFYBACKEND
AmplifyUIBuilder	amplifyuibuilder	AWS_ENDPOINT_URL_AMPLIFYUIBUILDER
API Gateway	apigateway	AWS_ENDPOINT_URL_API_GATEWAY
ApiGatewayManagementApi	apigatewaymanagementapi	AWS_ENDPOINT_URL_APIGATEWAYMANAGEMENTAPI
ApiGatewayV2	apigatewayv2	AWS_ENDPOINT_URL_APIGATEWAYV2
AppConfig	appconfig	AWS_ENDPOINT_URL_APPCONFIG
AppConfigData	appconfigdata	AWS_ENDPOINT_URL_APPCONFIGDATA
Appflow	appflow	AWS_ENDPOINT_URL_APPFLOW
AppIntegrations	appintegrations	AWS_ENDPOINT_URL_APPINTEGRATIONS
Application Auto Scaling	application_auto_scaling	AWS_ENDPOINT_URL_APPLICATION_AUTO_SCALING
Application Insights	application_insights	AWS_ENDPOINT_URL_APPLICATION_INSIGHTS
ApplicationCostProfiler	applicationcostprofiler	AWS_ENDPOINT_URL_APPLICATIONCOSTPROFILER
App Mesh	appmesh	AWS_ENDPOINT_URL_APP_MESH
AppRunner	apprunner	AWS_ENDPOINT_URL_APPRUNNER
AppStream	appstream	AWS_ENDPOINT_URL_APPSTREAM

serviceId	共享的服务标识密钥 AW**co**件	**AWS_ENDPOINT_URL_<SERVICE>** 环境变量
AppSync	appsync	AWS_ENDPOINT_URL_APPSYNC
Athena	athena	AWS_ENDPOINT_URL_ATHENA
AuditManager	auditmanager	AWS_ENDPOINT_URL_AUDITMANAGER
Auto Scaling	autoscaling	AWS_ENDPOINT_URL_AUTO_SCALING
Auto Scaling Plans	autoscaling-plans	AWS_ENDPOINT_URL_AUTO_SCALING_PLANS
Backup	backup	AWS_ENDPOINT_URL_BACKUP
Backup Gateway	backup-gateway	AWS_ENDPOINT_URL_BACKUP_GATEWAY
BackupStorage	backupstorage	AWS_ENDPOINT_URL_BACKUPSTORAGE
Batch	batch	AWS_ENDPOINT_URL_BATCH
billingconductor	billingconductor	AWS_ENDPOINT_URL_BILLINGCONDUCTOR
Braket	braket	AWS_ENDPOINT_URL_BRAKET
Budgets	budgets	AWS_ENDPOINT_URL_BUDGETS
Cost Explorer	cost-explorer	AWS_ENDPOINT_URL_COST_EXPLORER
Chime	chime	AWS_ENDPOINT_URL_CHIME
Chime SDK Identity	chime-sdk-identity	AWS_ENDPOINT_URL_CHIME_SDK_IDENTITY
Chime SDK Media Pipelines	chime-sdk-media-pipelines	AWS_ENDPOINT_URL_CHIME_SDK_MEDIA_PIPELINES
Chime SDK Meetings	chime-sdk-meetings	AWS_ENDPOINT_URL_CHIME_SDK_MEETINGS
Chime SDK Messaging	chime-sdk-messaging	AWS_ENDPOINT_URL_CHIME_SDK_MESSAGING
Cloud9	cloud9	AWS_ENDPOINT_URL_CLOUD9
CloudControl	cloudcontrol	AWS_ENDPOINT_URL_CLOUDCONTROL
CloudDirectory	clouddirectory	AWS_ENDPOINT_URL_CLOUDDIRECTORY
CloudFormation	cloudformation	AWS_ENDPOINT_URL_CLOUDFORMATION
CloudFront	cloudfront	AWS_ENDPOINT_URL_CLOUDFRONT
CloudHSM	cloudhsm	AWS_ENDPOINT_URL_CLOUDHSM

serviceId	共享的服务标识密钥 AWS **config** 文件	**AWS_ENDPOINT_URL_<SERVICE>** 环境变量
CloudHSM V2	cloudhsmv2	AWS_ENDPOINT_URL_CLOUDHSM_V2
CloudSearch	cloudsearch	AWS_ENDPOINT_URL_CLOUDSEARCH
CloudSearch Domain	cloudsearchdomain	AWS_ENDPOINT_URL_CLOUDSEARCH_DOMAIN
CloudTrail	cloudtrail	AWS_ENDPOINT_URL_CLOUDTRAIL
CloudWatch	cloudwatch	AWS_ENDPOINT_URL_CLOUDWATCH
codeartifact	codeartifact	AWS_ENDPOINT_URL_CODEARTIFACT
CodeBuild	codebuild	AWS_ENDPOINT_URL_CODEBUILD
CodeCommit	codecommit	AWS_ENDPOINT_URL_CODECOMMIT
CodeDeploy	codedeploy	AWS_ENDPOINT_URL_CODEDEPLOY
CodeGuru Reviewer	codeguru-reviewer	AWS_ENDPOINT_URL_CODEGURU_REVIEWER
CodeGuruProfiler	codeguruprofiler	AWS_ENDPOINT_URL_CODEGURUPROFILER
CodePipeline	codepipeline	AWS_ENDPOINT_URL_CODEPIPELINE
CodeStar	codestar	AWS_ENDPOINT_URL_CODESTAR
CodeStar connections	codestar-connections	AWS_ENDPOINT_URL_CODESTAR_CONNECTIONS
codestar notifications	codestar-notifications	AWS_ENDPOINT_URL_CODESTAR_NOTIFICATIONS
Cognito Identity	cognito-identity	AWS_ENDPOINT_URL_COGNITO_IDENTITY
Cognito Identity Provider	cognito-identity-provider	AWS_ENDPOINT_URL_COGNITO_IDENTITY_PROVIDER
Cognito Sync	cognito-sync	AWS_ENDPOINT_URL_COGNITO_SYNC
Comprehend	comprehend	AWS_ENDPOINT_URL_COMPREHEND
ComprehendMedical	comprehendmedical	AWS_ENDPOINT_URL_COMPREHENDMEDICAL
Compute Optimizer	compute-optimizer	AWS_ENDPOINT_URL_COMPUTE_OPTIMIZER
Config Service	config-service	AWS_ENDPOINT_URL_CONFIG_SERVICE
Connect	connect	AWS_ENDPOINT_URL_CONNECT
Connect Contact Lens	connect-contact-lens	AWS_ENDPOINT_URL_CONNECT_CONTACT_LENS

serviceId	共享的服务标识密钥 AWS **config** 文件	**AWS_ENDPOINT_URL_<SERVICE>** 环境变量
ConnectCampaigns	connect_campaigns	AWS_ENDPOINT_URL_CONNECTCAMPAIGNS
ConnectCases	connectcases	AWS_ENDPOINT_URL_CONNECTCASES
ConnectParticipant	connectparticipant	AWS_ENDPOINT_URL_CONNECTPARTICIPANT
ControlTower	controltower	AWS_ENDPOINT_URL_CONTROLTOWER
Cost and Usage Report Service	cost_and_usage_report_service	AWS_ENDPOINT_URL_COST_AND_USAGE_REPORT_SERVICE
Customer Profiles	customer_profiles	AWS_ENDPOINT_URL_CUSTOMER_PROFILES
DataBrew	databrew	AWS_ENDPOINT_URL_DATABREW
DataExchange	dataexchange	AWS_ENDPOINT_URL_DATAEXCHANGE
Data Pipeline	datapipeline	AWS_ENDPOINT_URL_DATA_PIPELINE
DataSync	datasync	AWS_ENDPOINT_URL_DATASYNC
DAX	dax	AWS_ENDPOINT_URL_DAX
Detective	detective	AWS_ENDPOINT_URL_DETECTIVE
Device Farm	devicefarm	AWS_ENDPOINT_URL_DEVICE_FARM
DevOps Guru	devops_guru	AWS_ENDPOINT_URL_DEVOPS_GURU
Direct Connect	directconnect	AWS_ENDPOINT_URL_DIRECT_CONNECT
Application Discovery Service	application_discovery_service	AWS_ENDPOINT_URL_APPLICATION_DISCOVERY_SERVICE
DLM	dlm	AWS_ENDPOINT_URL_DLM
Database Migration Service	database_migration_service	AWS_ENDPOINT_URL_DATABASE_MIGRATION_SERVICE
DocDB	docdb	AWS_ENDPOINT_URL_DOCDB
drs	drs	AWS_ENDPOINT_URL_DRS
Directory Service	directory_service	AWS_ENDPOINT_URL_DIRECTORY_SERVICE
DynamoDB	dynamodb	AWS_ENDPOINT_URL_DYNAMODB
DynamoDB Streams	dynamodb_streams	AWS_ENDPOINT_URL_DYNAMODB_STREAMS

serviceId	共享的服务标识密钥 AWS **config** 文件	**AWS_ENDPOINT_URL_<SERVICE>** 环境变量
EBS	ebs	AWS_ENDPOINT_URL_EBS
EC2	ec2	AWS_ENDPOINT_URL_EC2
EC2 Instance Connect	ec2_instance_connect	AWS_ENDPOINT_URL_EC2_INSTANCE_CONNECT
ECR	ecr	AWS_ENDPOINT_URL_ECR
ECR PUBLIC	ecr_public	AWS_ENDPOINT_URL_ECR_PUBLIC
ECS	ecs	AWS_ENDPOINT_URL_ECS
EFS	efs	AWS_ENDPOINT_URL_EFS
EKS	eks	AWS_ENDPOINT_URL_EKS
Elastic Inference	elastic_inference	AWS_ENDPOINT_URL_ELASTIC_INFERENCE
ElastiCache	elasticache	AWS_ENDPOINT_URL_ELASTICACHE
Elastic Beanstalk	elastic_beanstalk	AWS_ENDPOINT_URL_ELASTIC_BEANSTALK
Elastic Transcoder	elastic_transcoder	AWS_ENDPOINT_URL_ELASTIC_TRANSCODER
Elastic Load Balancing	elastic_load_balancing	AWS_ENDPOINT_URL_ELASTIC_LOAD_BALANCING
Elastic Load Balancing v2	elastic_load_balancing_v2	AWS_ENDPOINT_URL_ELASTIC_LOAD_BALANCING_V2
EMR	emr	AWS_ENDPOINT_URL_EMR
EMR containers	emr_containers	AWS_ENDPOINT_URL_EMR_CONTAINERS
EMR Serverless	emr_serverless	AWS_ENDPOINT_URL_EMR_SERVERLESS
Elasticsearch Service	es	AWS_ENDPOINT_URL_ELASTICSEARCH_SERVICE
EventBridge	eventbridge	AWS_ENDPOINT_URL_EVENTBRIDGE
Evidently	evidently	AWS_ENDPOINT_URL_EVIDENTLY
finspace	finspace	AWS_ENDPOINT_URL_FINSPACE
finspace data	finspace_data	AWS_ENDPOINT_URL_FINSPACE_DATA
Firehose	firehose	AWS_ENDPOINT_URL_FIREHOSE
fis	fis	AWS_ENDPOINT_URL_FIS

serviceId	共享的服务标识密钥 AWS **config** 文件	**AWS_ENDPOINT_URL_<SERVICE>** 环境变量
FMS	fms	AWS_ENDPOINT_URL_FMS
forecast	forecast	AWS_ENDPOINT_URL_FORECAST
forecastquery	forecastquery	AWS_ENDPOINT_URL_FORECASTQUERY
FraudDetector	frauddetector	AWS_ENDPOINT_URL_FRAUDDETECTOR
FSx	fsx	AWS_ENDPOINT_URL_FSX
GameLift	gamelift	AWS_ENDPOINT_URL_GAMELIFT
GameSparks	gamesparks	AWS_ENDPOINT_URL_GAMESPARKS
Glacier	glacier	AWS_ENDPOINT_URL_GLACIER
Global Accelerator	globalaccelerator	AWS_ENDPOINT_URL_GLOBAL_ACCELERATOR
Glue	glue	AWS_ENDPOINT_URL_GLUE
grafana	grafana	AWS_ENDPOINT_URL_GRAFANA
Greengrass	greengrass	AWS_ENDPOINT_URL_GREENGRASS
GreengrassV2	greengrassv2	AWS_ENDPOINT_URL_GREENGRASSV2
GroundStation	groundstation	AWS_ENDPOINT_URL_GROUNDSTATION
GuardDuty	guardduty	AWS_ENDPOINT_URL_GUARDDUTY
Health	health	AWS_ENDPOINT_URL_HEALTH
HealthLake	healthlake	AWS_ENDPOINT_URL_HEALTHLAKE
Honeycode	honeycode	AWS_ENDPOINT_URL_HONEYCODE
IAM	iam	AWS_ENDPOINT_URL_IAM
identitystore	identitystore	AWS_ENDPOINT_URL_IDENTITYSTORE
imagebuilder	imagebuilder	AWS_ENDPOINT_URL_IMAGEBUILDER
ImportExport	importexport	AWS_ENDPOINT_URL_IMPORTEXPORT
Inspector	inspector	AWS_ENDPOINT_URL_INSPECTOR
Inspector2	inspector2	AWS_ENDPOINT_URL_INSPECTOR2
IoT	iot	AWS_ENDPOINT_URL_IOT

serviceId	共享的服务标识密钥 AWS **config** 文件	**AWS_ENDPOINT_URL_<SERVICE>** 环境变量
IoT Data Plane	iot_data_plane	AWS_ENDPOINT_URL_IOT_DATA_PLANE
IoT Jobs Data Plane	iot_jobs_data_plane	AWS_ENDPOINT_URL_IOT_JOBS_DATA_PLANE
IoT 1Click Devices Service	iot_1click_devices_service	AWS_ENDPOINT_URL_IOT_1CLICK_DEVICES_SERVICE
IoT 1Click Projects	iot_1click_projects	AWS_ENDPOINT_URL_IOT_1CLICK_PROJECTS
IoTAnalytics	iotanalytics	AWS_ENDPOINT_URL_IOTANALYTICS
IotDeviceAdvisor	iotdeviceadvisor	AWS_ENDPOINT_URL_IOTDEVICEADVISOR
IoT Events	iot_events	AWS_ENDPOINT_URL_IOT_EVENTS
IoT Events Data	iot_events_data	AWS_ENDPOINT_URL_IOT_EVENTS_DATA
IoTFleetHub	iotfleethub	AWS_ENDPOINT_URL_IOTFLEETHUB
IoTFleetWise	iotfleetwise	AWS_ENDPOINT_URL_IOTFLEETWISE
IoTSecureTunneling	iotsecuretunneling	AWS_ENDPOINT_URL_IOTSECURETUNNELING
IoTSiteWise	iotsitewise	AWS_ENDPOINT_URL_IOTSITEWISE
IoTThingsGraph	iotthingsgraph	AWS_ENDPOINT_URL_IOTTHINGSGRAPH
IoTTwinMaker	iottwinmaker	AWS_ENDPOINT_URL_IOTTWINMAKER
IoT Wireless	iot_wireless	AWS_ENDPOINT_URL_IOT_WIRELESS
ivs	ivs	AWS_ENDPOINT_URL_IVS
ivschat	ivschat	AWS_ENDPOINT_URL_IVSCHAT
Kafka	kafka	AWS_ENDPOINT_URL_KAFKA
KafkaConnect	kafkaconnect	AWS_ENDPOINT_URL_KAFKACONNECT
kendra	kendra	AWS_ENDPOINT_URL_KENDRA
Keyspaces	keyspaces	AWS_ENDPOINT_URL_KEYSPACES
Kinesis	kinesis	AWS_ENDPOINT_URL_KINESIS
Kinesis Video Archived Media	kinesis_video_archived_media	AWS_ENDPOINT_URL_KINESIS_VIDEO_ARCHIVED_MEDIA

serviceId	共享的服务标识密钥 AWS **config** 文件	**AWS_ENDPOINT_URL_<SERVICE>** 环境变量
Kinesis Video Media	kinesis_video_media	AWS_ENDPOINT_URL_KINESIS_VIDEO_MEDIA
Kinesis Video Signaling	kinesis_video_signaling	AWS_ENDPOINT_URL_KINESIS_VIDEO_SIGNALING
Kinesis Analytics	kinesis_analytics	AWS_ENDPOINT_URL_KINESIS_ANALYTICS
Kinesis Analytics V2	kinesis_analytics_v2	AWS_ENDPOINT_URL_KINESIS_ANALYTICS_V2
Kinesis Video	kinesis_video	AWS_ENDPOINT_URL_KINESIS_VIDEO
KMS	kms	AWS_ENDPOINT_URL_KMS
LakeFormation	lakeformation	AWS_ENDPOINT_URL_LAKEFORMATION
Lambda	lambda	AWS_ENDPOINT_URL_LAMBDA
Lex Model Building Service	lex_model_building_service	AWS_ENDPOINT_URL_LEX_MODEL_BUILDING_SERVICE
Lex Runtime Service	lex_runtime_service	AWS_ENDPOINT_URL_LEX_RUNTIME_SERVICE
Lex Models V2	lex_models_v2	AWS_ENDPOINT_URL_LEX_MODELS_V2
Lex Runtime V2	lex_runtime_v2	AWS_ENDPOINT_URL_LEX_RUNTIME_V2
License Manager	license_manager	AWS_ENDPOINT_URL_LICENSE_MANAGER
License Manager User Subscriptions	license_manager_user_subscriptions	AWS_ENDPOINT_URL_LICENSE_MANAGER_USER_SUBSCRIPTIONS
Lightsail	lightsail	AWS_ENDPOINT_URL_LIGHTSAIL
Location	location	AWS_ENDPOINT_URL_LOCATION
CloudWatch Logs	cloudwatch_logs	AWS_ENDPOINT_URL_CLOUDWATCH_LOGS
LookoutEquipment	lookoutequipment	AWS_ENDPOINT_URL_LOOKOUTEQUIPMENT
LookoutMetrics	lookoutmetrics	AWS_ENDPOINT_URL_LOOKOUTMETRICS
LookoutVision	lookoutvision	AWS_ENDPOINT_URL_LOOKOUTVISION
m2	m2	AWS_ENDPOINT_URL_M2
Machine Learning	machine_learning	AWS_ENDPOINT_URL_MACHINE_LEARNING
Macie	macie	AWS_ENDPOINT_URL_MACIE

serviceId	共享的服务标识密钥 AWS **config** 文件	**AWS_ENDPOINT_URL_<SERVICE>** 环境变量
Macie2	macie2	AWS_ENDPOINT_URL_MACIE2
ManagedBlockchain	managedblockchain	AWS_ENDPOINT_URL_MANAGEDBLOCKCHAIN
Marketplace Catalog	marketplacecatalog	AWS_ENDPOINT_URL_MARKETPLACE_CATALOG
Marketplace Entitlement Service	marketplaceentitlementservice	AWS_ENDPOINT_URL_MARKETPLACE_ENTITLEMENT_SERVICE
Marketplace Commerce Analytics	marketplacecommerceanalytics	AWS_ENDPOINT_URL_MARKETPLACE_COMMERCE_ANALYTICS
MediaConnect	mediaconnect	AWS_ENDPOINT_URL_MEDIACONNECT
MediaConvert	mediaconvert	AWS_ENDPOINT_URL_MEDIACONVERT
MediaLive	medialive	AWS_ENDPOINT_URL_MEDIALIVE
MediaPackage	mediapackage	AWS_ENDPOINT_URL_MEDIAPACKAGE
MediaPackage Vod	mediapackagevod	AWS_ENDPOINT_URL_MEDIAPACKAGE_VOD
MediaStore	mediastore	AWS_ENDPOINT_URL_MEDIASTORE
MediaStore Data	mediastoredata	AWS_ENDPOINT_URL_MEDIASTORE_DATA
MediaTailor	mediatailor	AWS_ENDPOINT_URL_MEDIATAILOR
MemoryDB	memorydb	AWS_ENDPOINT_URL_MEMORYDB
Marketplace Metering	marketplacemetering	AWS_ENDPOINT_URL_MARKETPLACE_METERING
Migration Hub	migrationhub	AWS_ENDPOINT_URL_MIGRATION_HUB
mgn	mgn	AWS_ENDPOINT_URL_MGN
Migration Hub Refactor Spaces	migrationhub-refactor-spaces	AWS_ENDPOINT_URL_MIGRATION_HUB_REFACTOR_SPACES
MigrationHub Config	migrationhubconfig	AWS_ENDPOINT_URL_MIGRATIONHUB_CONFIG
MigrationHubOrchestrator	migrationhub-orchestrator	AWS_ENDPOINT_URL_MIGRATIONHUBORCHESTRATOR
MigrationHubStrategy	migrationhubstrategy	AWS_ENDPOINT_URL_MIGRATIONHUBSTRATEGY
Mobile	mobile	AWS_ENDPOINT_URL_MOBILE
mq	mq	AWS_ENDPOINT_URL_MQ

serviceId	共享的服务标识密钥 AWS **config** 文件	**AWS_ENDPOINT_URL_<SERVICE>** 环境变量
MTurk	mturk	AWS_ENDPOINT_URL_MTURK
MWAA	mwaa	AWS_ENDPOINT_URL_MWAA
Neptune	neptune	AWS_ENDPOINT_URL_NEPTUNE
Network Firewall	network_firewall	AWS_ENDPOINT_URL_NETWORK_FIREWALL
NetworkManager	networkmanager	AWS_ENDPOINT_URL_NETWORKMANAGER
nimble	nimble	AWS_ENDPOINT_URL_NIMBLE
OpenSearch	opensearch	AWS_ENDPOINT_URL_OPENSEARCH
OpsWorks	opsworks	AWS_ENDPOINT_URL_OPSWORKS
OpsWorksCM	opsworkscm	AWS_ENDPOINT_URL_OPSWORKSCM
Organizations	organizations	AWS_ENDPOINT_URL_ORGANIZATIONS
Outposts	outposts	AWS_ENDPOINT_URL_OUTPOSTS
Panorama	panorama	AWS_ENDPOINT_URL_PANORAMA
Personalize	personalize	AWS_ENDPOINT_URL_PERSONALIZE
Personalize Events	personalize_events	AWS_ENDPOINT_URL_PERSONALIZE_EVENTS
Personalize Runtime	personalize_runtime	AWS_ENDPOINT_URL_PERSONALIZE_RUNTIME
PI	pi	AWS_ENDPOINT_URL_PI
Pinpoint	pinpoint	AWS_ENDPOINT_URL_PINPOINT
Pinpoint Email	pinpoint_email	AWS_ENDPOINT_URL_PINPOINT_EMAIL
Pinpoint SMS Voice	pinpoint_sms_voice	AWS_ENDPOINT_URL_PINPOINT_SMS_VOICE
Pinpoint SMS Voice V2	pinpoint_sms_voice_v2	AWS_ENDPOINT_URL_PINPOINT_SMS_VOICE_V2
Polly	polly	AWS_ENDPOINT_URL_POLLY
Pricing	pricing	AWS_ENDPOINT_URL_PRICING
PrivateNetworks	privatenetworks	AWS_ENDPOINT_URL_PRIVATENETWORKS
Proton	proton	AWS_ENDPOINT_URL_PROTON
QLDB	qldb	AWS_ENDPOINT_URL_QLDB

serviceId	共享的服务标识密钥 AW**co**件	**AWS_ENDPOINT_URL_<SERVICE>** 环境变量
QLDB Session	qldb-session	AWS_ENDPOINT_URL_QLDB_SESSION
QuickSight	quicksight	AWS_ENDPOINT_URL_QUICKSIGHT
RAM	ram	AWS_ENDPOINT_URL_RAM
rbin	rbin	AWS_ENDPOINT_URL_RBIN
RDS	rds	AWS_ENDPOINT_URL_RDS
RDS Data	rds-data	AWS_ENDPOINT_URL_RDS_DATA
Redshift	redshift	AWS_ENDPOINT_URL_REDSHIFT
Redshift Data	redshift-data	AWS_ENDPOINT_URL_REDSHIFT_DATA
Redshift Serverless	redshift-serverless	AWS_ENDPOINT_URL_REDSHIFT_SERVERLESS
Rekognition	rekognition	AWS_ENDPOINT_URL_REKOGNITION
resiliencehub	resiliencehub	AWS_ENDPOINT_URL_RESILIENCEHUB
Resource Groups	resource-groups	AWS_ENDPOINT_URL_RESOURCE_GROUPS
Resource Groups Tagging API	resourcegroupstaggingapi	AWS_ENDPOINT_URL_RESOURCE_GROUPS_TAGGING_API
RoboMaker	robomaker	AWS_ENDPOINT_URL_ROBOMAKER
RolesAnywhere	rolesanywhere	AWS_ENDPOINT_URL_ROLESANYWHERE
Route 53	route-53	AWS_ENDPOINT_URL_ROUTE_53
Route53 Recovery Cluster	route53-recovery-cluster	AWS_ENDPOINT_URL_ROUTE53_RECOVERY_CLUSTER
Route53 Recovery Control Config	route53-recovery-control-config	AWS_ENDPOINT_URL_ROUTE53_RECOVERY_CONTROL_CONFIG
Route53 Recovery Readiness	route53-recovery-readiness	AWS_ENDPOINT_URL_ROUTE53_RECOVERY_READINESS
Route 53 Domains	route-53-domains	AWS_ENDPOINT_URL_ROUTE_53_DOMAINS
Route53Resolver	route53resolver	AWS_ENDPOINT_URL_ROUTE53RESOLVER
RUM	rum	AWS_ENDPOINT_URL_RUM
S3	s3	AWS_ENDPOINT_URL_S3

serviceId	共享的服务标识密钥 AWS **config** 文件	**AWS_ENDPOINT_URL_<SERVICE>** 环境变量
S3 Control	s3_control	AWS_ENDPOINT_URL_S3_CONTROL
S3Outposts	s3_outposts	AWS_ENDPOINT_URL_S3OUTPOSTS
SageMaker	sagemaker	AWS_ENDPOINT_URL_SAGEMAKER
SageMaker A2I Runtime	sagemaker_a2i_runtime	AWS_ENDPOINT_URL_SAGEMAKER_A2I_RUNTIME
Sagemaker Edge	sagemaker_edge	AWS_ENDPOINT_URL_SAGEMAKER_EDGE
SageMaker FeatureStore Runtime	sagemaker_featurestore_runtime	AWS_ENDPOINT_URL_SAGEMAKER_FEATURESTORE_RUNTIME
SageMaker Runtime	sagemaker_runtime	AWS_ENDPOINT_URL_SAGEMAKER_RUNTIME
savingsplans	savingsplans	AWS_ENDPOINT_URL_SAVINGSPLANS
schemas	schemas	AWS_ENDPOINT_URL_SCHEMAS
SimpleDB	simpledb	AWS_ENDPOINT_URL_SIMPLEDB
Secrets Manager	secrets_manager	AWS_ENDPOINT_URL_SECRETS_MANAGER
SecurityHub	securityhub	AWS_ENDPOINT_URL_SECURITYHUB
ServerlessApplicationRepository	serverlessapplicationrepository	AWS_ENDPOINT_URL_SERVERLESSAPPLICATIONREPOSITORY
Service Quotas	service_quotas	AWS_ENDPOINT_URL_SERVICE_QUOTAS
Service Catalog	service_catalog	AWS_ENDPOINT_URL_SERVICE_CATALOG
Service Catalog AppRegistry	service_catalog_appregistry	AWS_ENDPOINT_URL_SERVICE_CATALOG_APPREGISTRY
ServiceDiscovery	servicediscovery	AWS_ENDPOINT_URL_SERVICEDISCOVERY
SES	ses	AWS_ENDPOINT_URL_SES
SESv2	sesv2	AWS_ENDPOINT_URL_SESV2
Shield	shield	AWS_ENDPOINT_URL_SHIELD
signer	signer	AWS_ENDPOINT_URL_SIGNER
SMS	sms	AWS_ENDPOINT_URL_SMS
Snow Device Management	snow_device_management	AWS_ENDPOINT_URL_SNOW_DEVICE_MANAGEMENT

serviceId	共享的服务标识密钥 AWS **config** 文件	**AWS_ENDPOINT_URL_<SERVICE>** 环境变量
Snowball	snowball	AWS_ENDPOINT_URL_SNOWBALL
SNS	sns	AWS_ENDPOINT_URL_SNS
SQS	sqs	AWS_ENDPOINT_URL_SQS
SSM	ssm	AWS_ENDPOINT_URL_SSM
SSM Contacts	ssm_contacts	AWS_ENDPOINT_URL_SSM_CONTACTS
SSM Incidents	ssm_incidents	AWS_ENDPOINT_URL_SSM_INCIDENTS
SSO	sso	AWS_ENDPOINT_URL_SSO
SSO Admin	sso_admin	AWS_ENDPOINT_URL_SSO_ADMIN
SSO OIDC	sso_oidc	AWS_ENDPOINT_URL_SSO_OIDC
SFN	sfn	AWS_ENDPOINT_URL_SFN
Storage Gateway	storage_gateway	AWS_ENDPOINT_URL_STORAGE_GATEWAY
STS	sts	AWS_ENDPOINT_URL_STS
Support	support	AWS_ENDPOINT_URL_SUPPORT
Support App	support_app	AWS_ENDPOINT_URL_SUPPORT_APP
SWF	swf	AWS_ENDPOINT_URL_SWF
synthetics	synthetics	AWS_ENDPOINT_URL_SYNTHETICS
Textract	textract	AWS_ENDPOINT_URL_TEXTRACT
Timestream Query	timestream_query	AWS_ENDPOINT_URL_TIMESTREAM_QUERY
Timestream Write	timestream_write	AWS_ENDPOINT_URL_TIMESTREAM_WRITE
Transcribe	transcribe	AWS_ENDPOINT_URL_TRANSCRIBE
Transfer	transfer	AWS_ENDPOINT_URL_TRANSFER
Translate	translate	AWS_ENDPOINT_URL_TRANSLATE
Voice ID	voice_id	AWS_ENDPOINT_URL_VOICE_ID
WAF	waf	AWS_ENDPOINT_URL_WAF
WAF Regional	waf_regional	AWS_ENDPOINT_URL_WAF_REGIONAL

serviceId	共享的服务标识密钥AWS **config**文件	**AWS_ENDPOINT_URL_<SERVICE>** 环境变量
WAFV2	wafv2	AWS_ENDPOINT_URL_WAFV2
WellArchitected	wellarchitected	AWS_ENDPOINT_URL_WELLARCHITECTED
Wisdom	wisdom	AWS_ENDPOINT_URL_WISDOM
WorkDocs	workdocs	AWS_ENDPOINT_URL_WORKDOCS
WorkLink	worklink	AWS_ENDPOINT_URL_WORKLINK
WorkMail	workmail	AWS_ENDPOINT_URL_WORKMAIL
WorkMailMessageFlow	workmailmessageflow	AWS_ENDPOINT_URL_WORKMAILMESSAGEFLOW
WorkSpaces	workspaces	AWS_ENDPOINT_URL_WORKSPACES
WorkSpaces Web	workspaces_web	AWS_ENDPOINT_URL_WORKSPACES_WEB
XRay	xray	AWS_ENDPOINT_URL_XRAY

智能配置默认值

借助智能配置默认功能，AWSSDK 可以为其他配置设置提供预定义的、经过优化的默认值。

使用以下方法配置此功能：

defaults_mode-共享AWS **config**文件设置, **AWS_DEFAULTS_MODE**-环境变量

使用此设置，您可以选择与您的应用程序架构相匹配的模式，然后为您的应用程序提供优化的默认值。如果AWSSDK 设置具有明确设置的值，则该值始终优先。如果AWSSDK 设置没有明确设置值，并且defaults_mode不等于旧版，则此功能可以为针对您的应用程序优化的各种设置提供不同的默认值。设置可能包括以下内容：HTTP 通信设置、重试行为、服务区域端点设置，可能还包括任何与 SDK 相关的配置。使用此功能的客户可以获得针对常见使用场景量身定制的新配置默认值。如果你的defaults_mode不等于legacy，我们建议您在升级 SDK 时对应用程序进行测试，因为提供的默认值可能会随着最佳实践的变化而改变。

默认值：legacy

注意：新的主要版本的 SDK 将默认为standard。

有效值：

- legacy— 提供默认设置，这些设置因 SDK 而异，并且在建立之前就已存在defaults_mode。
- standard— 提供最新的推荐默认值，这些默认值在大多数情况下都应该可以安全运行。
- in-region— 基于标准模式构建，包括为调用的应用程序量身定制的优化AWS 服务从同一个地方出发AWS 区域。

- `cross-region`— 基于标准模式构建，包括为调用的应用程序量身定制的优化AWS 服务在不同的区域。
- `mobile`— 基于标准模式构建，包括为移动应用程序量身定制的优化。
- `auto`— 基于标准模式构建，包括实验功能。SDK 会尝试发现运行时环境以自动确定适当的设置。自动检测是基于启发式的，无法提供 100% 的准确性。如果无法确定运行时环境，`standard`使用模式。自动检测可能会查询实例元数据和用户数据，这可能会带来延迟。如果启动延迟对您的应用程序至关重要，我们建议您选择明确的`defaults_mode`相反。

在中设置此值的示例`config`文件:

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

以下参数可能会根据选择进行优化`defaults_mode`:

- `retryMode`— 指定 SDK 如何尝试重试。请参阅重试行为 (p. 50)。
- `stsRegionalEndpoints`— 指定 SDK 如何确定AWS 服务它用来与之交谈的端点AWS Security Token Service(AWS STS)。请参阅AWS STS区域化终端节点 (p. 43)。
- `s3UsEast1RegionalEndpoints`— 指定 SDK 如何确定AWS用于与 Amazon S3 通信的服务终端节点`us-east-1`区域。
- `connectTimeoutInMillis`— 在套接字上进行初始连接尝试后，超时之前的时间。如果客户端没有收到连接握手完成的消息，则客户端会放弃并使操作失败。
- `tlsNegotiationTimeoutInMillis`— 从发送 CLIENT HELLO 消息到客户端和服务器完全协商密码并交换密钥，TLS 握手可能花费的最大时间。

每个设置的默认值会根据不同而变化`defaults_mode`为您的应用程序选择。这些值目前设置如下（可能会发生变化）：

参数	**`standard`模式**	**`in-region`模式**	**`cross-region`模式**	**`mobile`模式**
`retryMode`	`standard`	`standard`	`standard`	`standard`
`stsRegionalEndpoints`	`regional`	`regional`	`regional`	`regional`
`s3UsEast1RegionalEndpoints`	`regional`	`regional`	`regional`	`regional`
`connectTimeoutInMillis`	3100	1100	3100	30000
`tlsNegotiationTimeoutInMillis`	3100	1100	3100	30000

例如，如果`defaults_mode`你选择的是`standard`，然后是的值`standard`将被分配给`retry_mode`（从有效的`retry_mode`选项）和的值`regional`将被分配给`stsRegionalEndpoints`（从有效的`stsRegionalEndpoints`选项）。

兼容AWSSDKS

以下 SDK 支持本页上描述的功能和设置，但请注意部分例外情况：

SDK	支持	备注或更多信息
AWS CLI v2	否	

SDK	支持	备注或更多信息
C++ 软件开发工具包	是	参数未优化：stsRegionalEndpoints，s3UsEast1Re
适用于 Go V2 的 SDK (1.x)	是	参数未优化：retryMode，stsRegionalEndpoints，
适用于 Go 1.x 的 SDK (V1)	否	
适用于 Java 的 SDK 2.x	是	参数未优化：stsRegionalEndpoints。
适用于 Java 的 SDK 1.x	否	
适用于JavaScript3.x	是	参数未优化：stsRegionalEndpoints，s3UsEast1Re 称为connectionTimeout。
适用于JavaScript2.x	否	
适用于.NET 3.x 的开发工具包	是	参数未优化：connectTimeoutInMillis，tlsNegoti
适用于 PHP 的 SDK 3.x	是	参数未优化：tlsNegotiationTimeoutInMillis。
适用于 Python 的 SDK (Boto3)	是	参数未优化：tlsNegotiationTimeoutInMillis。
适用于 Ruby 3.x 的 SDK	是	

AWS通用运行时 (CRT) 库

的AWS通用运行时 (CRT) 库是 SDK 的基础库。CRT 是一个由独立封装组成的模块化系列，用 C 语言编写。每个封装都为不同的所需功能提供了良好的性能和最小的占用空间。这些功能在所有 SDK 中都是通用的，可提供更好的代码重用、优化和准确性。这些套餐是：

- awslabs/aws-c-auth:AWS客户端身份验证（标准凭据提供程序和签名 (sigv4)）
- awslabs/aws-c-cal: 加密基元类型、哈希（MD5、SHA256、SHA256 HMAC）、签名者、AES
- awslabs/aws-c-common: 基本数据结构、线程/同步基元类型、缓冲区管理、stdlib 相关函数
- awslabs/aws-c-compression: 压缩算法（霍夫曼编码/解码）
- awslabs/aws-c-event-stream: 事件流消息处理（标头、前奏、有效载荷、crc/trailer）、事件流上的远程过程调用 (RPC) 实现
- awslabs/aws-c-http: C99 对 HTTP/1.1 和 HTTP/2 规范的实现
- awslabs/aws-c-io: 套接字（TCP、UDP）、DNS、管道、事件循环、通道、SSL/TLS
- awslabs/aws-c-iot: C99 的实施情况AWS物联网云服务与设备集成
- awslabs/aws-c-mqtt: 适用于物联网 (IoT) 的标准轻量级消息协议
- awslabs/aws-c-s3: 用于与 Amazon S3 服务通信的 C99 库实现，旨在最大限度地提高高带宽 Amazon EC2 实例的吞吐量
- awslabs/aws-c-sdkutils: 用于解析和管理的实用程序库AWS配置文件
- awslabs/aws-checksums: 跨平台硬件加速的 crc32c 和 CRC32，可回退到高效的软件实现
- awslabs/aws-lc: 由维护的通用密码库AWS密码学团队AWS以及他们的客户，基于谷歌 BoringSSL 项目和 OpenSSL 项目的代码
- awslabs/s2n: C99 实施 TLS/SSL 协议，该协议旨在小巧快速，将安全性放在首位

CRT 可在除 Go 之外的所有软件开发工具包中使用。

CRT 依赖关系

CRT 库构成了一个由关系和依赖关系组成的复杂网络。如果您需要直接从源代码构建 CRT，那么了解这些关系会很有帮助。但是，大多数用户通过其语言 SDK 访问 CRT 功能（例如AWSC++ 开发工具包或AWS适用于 Java 的软件开发工具包）或其语言 IoT 设备 SDK（例如AWS适用于 C++ 的 IoT SDAWS适用于 Java 的物联网开发工具包）在下图中，“语言 CRT 绑定” 框指的是封装特定语言 SDK 的 CRT 库的软件包。这是以下形式的软件包的集合`aws-crt-*`，其中 '*' 是一种 SDK 语言（例如aws-crt-cpp要么aws-crt-java）。

以下是 CRT 库的分层依赖关系的插图。

维护和支持

有关可以帮助您开发应用程序的工具的概述AWS，请参阅可供构建的工具AWS。有关支持的信息，请参阅AWS知识中心。

以下主题涵盖以下主题的维护和版本支持政策AWS软件开发工具包。

主题

AWSSDK 和工具维护政策

概览

本文档概述了以下各项的维护政策AWS软件开发套件 (SDK) 和工具，包括移动和 IoT SDK 及其底层依赖项。AWS定期提供AWS带有更新的 SDK 和工具，其中可能包含对新增或更新的支持AWSAPI、新功能、增强功能、错误修复、安全补丁或文档更新。更新还可以解决依赖关系、语言运行时和操作系统的变化。AWSSDK 版本发布给软件包管理器（例如 Maven、NuGet，PyPI），并可作为源代码获得GitHub。

我们建议用户留下来up-to-date使用 SDK 版本来跟上最新功能、安全更新和底层依赖关系。不建议继续使用不支持的 SDK 版本，这由用户自行决定。

版本控制

的AWSSDK 发布版本采用 X.Y.Z 的形式，其中 X 代表主要版本。增加 SDK 的主版本表明该 SDK 进行了重大而实质性的更改，以支持该语言中的新习语和模式。当公共接口（例如类、方法、类型等）、行为或语义发生变化时，就会引入主要版本。应用程序需要更新才能使用最新的 SDK 版本。请务必谨慎更新主要版本，并遵循提供的升级指南AWS。

SDK 主要版本生命周期

主要 SDK 和 Tools 版本的生命周期由 5 个阶段组成，概述如下。

- 开发者预览版（第 0 阶段）-在此阶段，不支持 SDK，不应在生产环境中使用，并且仅用于抢先体验和反馈目的。未来的版本可能会引入重大更改。曾经AWS将某个版本标识为稳定产品，它可能会将其标记为候选版本。除非出现重大错误，否则候选版本已准备就绪 GA 发布，并且将获得完整版本AWS支持。
- 正式上市 (GA)（第 1 阶段）-在此阶段，将完全支持 SDK。AWS将提供常规的 SDK 版本，其中包括对新服务的支持、现有服务的 API 更新以及错误和安全修复。对于工具，AWS将提供包含新功能更新和错误修复的常规版本。AWS将支持 GA 版本的 SDK 适用于至少 24 个月。
- 维护公告（第 2 阶段）- AWS将在 SDK 进入维护模式前至少 6 个月发布公告。在此期间，SDK 将继续得到全面支持。通常，维护模式是在下一个主要版本过渡到 GA 的同时宣布的。
- 维护（第 3 阶段）-在维护模式下，AWS将 SDK 版本限制为仅解决严重的错误修复和安全问题。SDK 不会收到新服务或现有服务的 API 更新，也不会更新以支持新区域。维护模式有一个默认期限为 12 个月，除非另有说明。
- 终止支持（第 4 阶段）-当 SDK 达到终止支持时，它将不再接收更新或版本。之前发布的版本将继续通过公共包管理器提供，并且代码将保持不变GitHub。的GitHub存储库可能已存档。使用已到达 e 的 SDKnd-of-support由用户自行决定。我们建议用户升级到新的主要版本。

以下是 SDK 主要版本生命周期的直观说明。请注意，下面显示的时间表仅供参考，不具约束力。

依赖生命周期

大多数AWSSDK 具有底层依赖关系，例如语言运行时、操作系统或第三方库和框架。这些依赖关系通常与语言社区或拥有该特定组件的供应商有关。每个社区或供应商都会发布自己的 end-of-support他们的产品时间表。

以下术语用于对潜在的第三方依赖关系进行分类：

- 操作系统 (OS)：例子包括亚马逊 Linux AMI、亚马逊 Linux 2、Windows 2008、Windows 2012、Windows 2016 等。
- 语言运行时间：示例包括 Java 7、Java 8、Java 11、.NET Core、.NET Standard、.NET PCL 等。
- 第三方库/框架：示例包括 OpenSSL、.NET Framework 4.5、Java EE 等。

我们的政策是在社区或供应商终止对 SDK 依赖项的支持后至少 6 个月内继续支持 SDK 依赖关系。但是，此策略可能会因具体的依赖关系而有所不同。

Note

AWS保留在不增加主要 SDK 版本的情况下停止对底层依赖项的支持的权利

沟通方式

维护公告通过多种方式传达：

- 我们会向受影响的账户发送一封电子邮件公告，宣布我们计划终止对特定 SDK 版本的支持。这封电子邮件将概述通往 e 的路径nd-of-support，指定活动时间表并提供升级指导。
- AWSSDK 文档，例如 API 参考文档、用户指南、SDK 产品营销页面以及GitHub自述文件已更新，以指明活动时间表，并提供有关升级受影响应用程序的指导。
- 一个AWS已发布博客文章，概述了通往 e 的道路nd-of-support，并重申了竞选时间表。
- 已在 SDK 中添加了弃用警告，概述了 e 的路径nd-of-support并链接到 SDK 文档。

要查看可用的主要版本列表AWSSDK 和工具以及它们在维护生命周期中所处的位置，请参阅the section called “版本支持矩阵” (p. 74)。

AWSSDK 和工具版本支持矩阵

下面的矩阵显示了可用列表AWSSDK 主要版本以及它们在维护生命周期中所处的位置，以及相关的时间表。有关软件开发套件 (SDK) 和工具的主要版本的生命周期及其底层依赖关系的详细信息，请参阅the section called “维护政策” (p. 73)

SDK	主要版本	当前阶段	正式上市日期	注意
AWS CLI	1.x	公开发行	9/2/2013	
AWS CLI	2.x	公开发行	2020 年 10 月 2 日	
C++ 软件开发工具包	1.x	公开发行	9/2/2015	
适用于 Go V2 的 SDK	V2 1.x	公开发行	1/19/2021	

SDK	主要版本	当前阶段	正式上市日期	注意
Go 版 SDK	1.x	公开发行	11/19/2015	
Java 版 SDK	1.x	公开发行	2010 年 3 月 25 日	
Java 版 SDK	2.x	公开发行	11/20/2018	
适用于JavaScript	1.x	支持终止	5/6/2013	
适用于JavaScript	2.x	公开发行	6/19/2014	
适用于JavaScript	3.x	公开发行	12/15/2020	
Kotlin 版 SDK	1.x	开发人员预览		
适用于.NET 的 S	1.x	支持终止	11/2009	
适用于.NET 的 S	2.x	支持终止	11/8/2013	
适用于.NET 的 S	3.x	公开发行	7/28/2015	
PHP 版 SDK	2.x	支持终止	11/2/2012	
PHP 版 SDK	3.x	公开发行	5/27/2015	
适用于 Python 的 SDK (Boto2)	1.x	支持终止	7/13/2011	
适用于 Python 的 SDK (Boto3)	1.x	公开发行	2015 年 6 月 22 日	
适用于 Python 的 SDK (Botocore	1.x	公开发行	2015 年 6 月 22 日	
Ruby 软件开发工具包	1.x	支持终止	7/14/2011	
Ruby 软件开发工具包	2.x	支持终止	2015 年 2 月 15 日	
Ruby 软件开发工具包	3.x	公开发行	8/29/2017	
Rust 版 SDK	1.x	开发人员预览		
适用于 Swift	1.x	开发人员预览		
适用于 PowerShell 的工具	2.x	支持终止	11/8/2013	
适用于 PowerShell 的工具	3.x	支持终止	7/29/2015	
适用于 PowerShell 的工具	4.x	公开发行	11/21/2019	

IDE 工具箱

AWS集成开发环境 (IDE) 工具包是允许访问的插件和扩展AWS来自你的 IDE 的服务。

有关每个 IDE 工具包的详细信息，请参阅以下 Toolkit 用户指南：

- AWS Toolkit for Visual Studio
- AWS Toolkit for Visual Studio Code
- AWS Toolkit for JetBrains

以下各节包含支持信息、维护报告和通知AWSIDE 工具包。

遥测通知

AWSIDE Toolkits 可以收集和存储客户端遥测数据，为未来决策提供信息AWS工具包发布。收集的数据量化了您对以下内容的使用情况AWS工具包。

要了解有关在所有地区收集的遥测数据的更多信息AWSIDE 工具包，请参阅通用定义.json文档在`aws-toolkit-common`Github 存储库

有关每个人收集的遥测数据的详细信息AWSIDE 工具包，参考以下资源文档AWS工具包的 Github 存储库：

- AWS Toolkit for Visual Studio
- AWS Toolkit for Visual Studio Code
- AWS Toolkit for JetBrains

肯定的AWS可通过工具包访问的服务可能会收集额外的客户端遥测数据，例如 AmazonCodeWhisperer。有关收集的数据类型的详细信息CodeWhisperer或者如何选择退出客户端遥测CodeWhisperer，请参阅与之共享您的数据AWS中的话题亚马逊CodeWhisperer用户指南。

的文档历史记录AWSSDK 和工具参考指南

下表描述了重要的新增内容和更新AWSSDK 和工具参考指南。如需对此文档更新的通知，您可以订阅 RSS 源。

变更	说明	日期
IAM 最佳实践更新 (p. 77)	更新了指南，使其符合 IAM 最佳实践。有关更多信息，请参阅 IAM 安全最佳实践。	2023 年 2 月 27 日
SSO 更新 (p. 77)	更新了新 SSO 令牌配置的 SSO 凭证。	2022年11月19日
设置更新 (p. 77)	更新了常规配置和 Amazon S3 多区域接入点的支持表。	2022 年 11 月 17 日
设置更新 (p. 77)	更新了 IMDS 客户端和 IMDS 凭证的清晰度。更新了环境变量。	2022 年 11 月 4 日
更新欢迎页面 (p. 77)	宣布亚马逊CodeWhisperer。	2022 年 9 月 22 日
更改单点登录的服务名称 (p. 77)	更新以反映这一点AWSSSO 现在被称为AWS IAM Identity Center。	2022 年 7 月 26 日
设置更新 (p. 77)	对配置文件详细信息和支持的设置进行了次要更新。	2022 年 6 月 15 日
更新 (p. 77)	本指南几乎所有部分都进行了大规模更新。	2022 年 2 月 1 日
首次发布 (p. 77)	本指南的第一版已向公众发布。	2020 年 3 月 13 日

AWS 词汇表

有关最新的 AWS 术语，请参阅《AWS 词汇表参考》中的 AWS 词汇表。

本文属于机器翻译版本。若本译文内容与英语原文存在差异，则一律以英文原文为准。