Amazon 一般参考

参考指南

版本 1.0

Amazon 一般参考: 参考指南

Amazon Web Services 文档中描述的 Amazon Web Services 服务或功能可能因区域而异。要查看适用于中国区域的差异，请参阅中国的 Amazon Web Services 服务入门。

Table of Contents

Amazon 一般参考

Amazon一般参考提供了亚马逊云科技的有用信息。

目录

Amazon 安全凭证

当您与 Amazon 交互时，可指定 Amazon 安全凭证 以验证您的身份以及您是否有权访问所请求的资源。Amazon 使用安全凭证来对您的请求进行身份验证和授权。

例如，如果要从 Amazon Simple Storage Service (Amazon S3) 存储桶下载受保护的文件，则您的凭证必须允许该访问。如果您的凭证无权下载该文件，Amazon 会拒绝您的请求。但是，下载公开共享的 Amazon S3 存储桶中的文件不需要您的 Amazon 安全凭证。

目录

Amazon Web Services 账户 根用户凭证与 IAM 用户凭证

Amazon 中有两种不同类型的用户。您是账户拥有者（根用户），或者是 Amazon Identity and Access Management（IAM）用户。根用户在创建 Amazon 账户时创建，IAM 用户由根用户或账户的 IAM 管理员创建。所有 Amazon 用户都具有安全凭证。

根用户凭证

账户拥有者的凭证允许完全访问账户中的所有资源。您无法使用 IAM 策略显式拒绝根用户访问资源。您只能使用 Amazon Organizations 服务控制策略（SCP）来限制根用户的权限。因此，我们建议您创建一个具有管理员权限的 IAM 用户，以用于日常 Amazon 任务并锁定根用户的访问密钥。

有一些特定任务仅限于 Amazon Web Services 账户 根用户。例如，只有根用户可以关闭您的账户。如果您需要执行需要根用户的任务，请使用根用户的电子邮件地址和密码登录 Amazon Web Services Management Console。有关更多信息，请参阅需要根用户凭证的任务 (p. 2)。

IAM 凭证

通过 IAM，您可以安全地控制用户对 Amazon Web Services 账户 中 Amazon 服务和资源的访问。例如，如果您需要管理员级别权限，则可以创建 IAM 用户，为该用户授予完全访问权限，然后使用这些凭证与 Amazon 交互。如果需要修改或撤销权限，您可以删除或修改与该 IAM 用户相关联的策略。

如果多个用户需要访问您的 Amazon Web Services 账户，您可以为每个用户创建唯一的凭证并定义哪些用户有权访问哪些资源。您不必共享凭证。例如，您可以创建对 Amazon Web Services 账户 中的资源具有只读访问权限的 IAM 用户，并将这些凭证分发给您的用户。

需要根用户凭证的任务

我们建议您使用具有适当权限的 IAM 用户来执行任务和访问 Amazon 资源。不过，您只能在以账户的根用户身份登录时才能执行下列任务。

任务

- 更改您的账户设置。这包括账户名称、电子邮件地址、根用户密码和根用户访问密钥。其他账户设置（例如联系人信息、付款货币偏好和区域）不需要根用户凭证。

- 恢复 IAM 用户权限。如果唯一的 IAM 管理员意外撤消了自己的权限，您可以使用根用户身份登录来编辑策略并还原这些权限。
- 激活 IAM 对账单和成本管理控制台的访问权限。
- 查看特定税务发票。具有 aws-portal:ViewBilling 权限的 IAM 用户还可以查看和下载 Amazon 欧洲的增值税发票，但不能查看和下载 Amazon Inc 或 Amazon Internet Services Pvt. Ltd (AISPL) 的增值税发票。
- 关闭 Amazon Web Services 账户。
- 更改 Amazon 支持计划或取消 Amazon 支持计划。有关更多信息，请参阅Amazon 的 IAM 支持。
- 已在预留实例 Marketplace 中注册为卖家。
- 为 S3 桶配置 MFA 删除。
- 编辑或删除一个包含无效 VPC ID 或 VPC 终端节点 ID 的 Amazon S3 存储桶策略。
- 注册 GovCloud。

问题排查

如果您无法使用您的根用户凭证完成以下任何任务，您的账户可能是 Amazon Organizations 中的组织的成员。如果组织管理员使用服务控制策略（SCP）来限制账户的权限，则您的根用户权限可能会受到影响。有关更多信息，请参阅《Amazon Organizations 用户指南》中的服务控制策略。

了解并获取您的 Amazon 凭证

Amazon 需要不同类型的安全凭证，具体取决于您访问 Amazon 的方式。例如，您需要用户名和密码才能登录 Amazon Web Services Management Console，并且需要访问密钥才能以编程方式调用 Amazon 或使用 Amazon Command Line Interface 或 Amazon Tools for PowerShell。

注意事项

- 请务必将以下内容保存在一个安全位置：与您的 Amazon Web Services 账户 关联的电子邮件地址、Amazon Web Services 账户 ID、根用户密码和账户访问密钥。如果您忘记或丢失了根用户密码，则必须有权访问与您的账户关联的电子邮件地址才能重置根用户密码。如果您忘记或丢失了访问密钥，则必须登录您的账户才能创建新访问密钥。
- 我们强烈建议您创建一个具有管理员权限的 IAM 用户，以用于日常 Amazon 任务并锁定根用户的密码和访问密钥。仅将根用户用于仅限于根用户的任务。
- 安全凭证特定于账户。如果您有权访问多个 Amazon 账户，则每个账户都必须有单独的凭证。
- 不要将您的 Amazon 凭证提供给第三方。

凭证

控制台访问

Amazon 中有两种不同类型的用户。您是账户拥有者（根用户），或者是 Amazon Identity and Access Management（IAM）用户。如何登录 Amazon Web Services Management Console 取决于您是根用户还是 IAM 用户。

目录

根用户电子邮件地址和密码

首次创建 Amazon Web Services 账户 时，您需要为该账户指定一个电子邮件地址，并为根用户指定一个密码。要以根用户身份登录您的 Amazon Web Services 账户，您需要提供此电子邮件地址和密码。根用户可以登录 Amazon Web Services Management Console 并使用 Security Credentials (安全凭证) 页面更改账户名称、电子邮件地址和密码。如果您忘记了根用户的密码，请打开控制台登录页面并选择 Forgot your password? (忘记了您的密码？) 来重置您的密码。此过程需要访问账户的电子邮件地址。

IAM 用户名和密码

IAM 用户由 Amazon Web Services 账户 中的根用户或 IAM 管理员创建。创建您的 IAM 用户的用户应向您提供账户别名或 12 位 Amazon Web Services 账户 ID、IAM 用户名和 IAM 用户的密码。IAM 用户可以使用控制台登录页面或以下登录 URL 登录，将 *account_id_or_alias* 替换为提供给您的账户别名或 Amazon Web Services 账户 ID：

```
https://account_id_or_alias.signin.aws.amazon.com/console/
```

如果您忘记了 IAM 用户的密码，请联系您的 IAM 管理员或账户拥有者。如果您的 IAM 管理员授予您管理自己 Amazon 凭证的权限，那么您可以使用 Security Credentials (安全凭证) 页面定期更改密码，这是安全最佳实践。

多重验证 (MFA)

多重身份验证 (MFA) 可以提供额外的安全级别，可以应用于您的 Amazon Web Services 账户。为了提高安全性，我们建议您对 Amazon Web Services 账户 根用户凭证和高权限 IAM 用户要求使用 MFA。有关更多信息，请参阅《IAM 用户指南》中的在 Amazon 中使用多重身份验证 (MFA) 。

启用 MFA 后，当您登录 Amazon Web Services 账户 时，系统会提示您输入用户名和密码，以及来自 MFA 设备的身份验证代码。添加 MFA 将为您的 Amazon Web Services 账户 设置和资源提供更高的安全保护。

默认情况下不启用 MFA (multi-factor authentication)。您可以前往 Security Credentials (安全凭证) 页面或 Amazon Web Services Management Console 中的 IAM 控制面板，为 Amazon Web Services 账户 根用户启用和管理 MFA 设备。有关为 IAM 用户启用 MFA 的更多信息，请参阅《IAM 用户指南》中的启用 MFA 设备。

以编程方式访问

您必须提供 Amazon 访问密钥才能以编程方式调用 Amazon 或使用 Amazon Command Line Interface 或 Amazon Tools for PowerShell。

创建访问密钥时，您将访问密钥 ID (例如 `AKIAIOSFODNN7EXAMPLE`) 和秘密访问密钥 (例如 `wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`) 创建为一组。秘密访问密钥仅在您创建它时可供下载。如果您没有下载秘密访问密钥或丢失了它，则必须创建新的秘密访问密钥。

您最多可以为每个用户 (根用户或 IAM 用户) 分配两个访问密钥。当您想要轮换访问密钥时，拥有两个访问密钥非常有用。当您禁用访问密钥时，您将无法使用它，但它会计入您的两个访问密钥限制。删除访问密钥后，意味着它将被永久删除且无法恢复，但可以替换为新的访问密钥。

以根用户身份登录时管理访问密钥

1. 以根用户身份登录Amazon Web Services Management Console 有关更多信息，请参阅《IAM 用户指南》中的以根用户身份登录。

2. 在右上角的导航栏中，选择您的账户名称或编号，然后选择 My Security Credentials (我的安全凭证)。
3. 展开 Access keys (access key ID and secret access key) 部分。
4. 请执行下列操作之一：

   - 要创建访问密钥，请选择 Create New Access Key (创建新的访问密钥)。如果您已经有两个访问密钥，则此按钮将被禁用，您必须先删除一个访问密钥，然后才能创建新的访问密钥。系统提示时，选择 Show Access Key (显示访问密钥) 或者 Download Key File (下载密钥文件)。这是您保存秘密访问密钥的唯一机会。将秘密访问密钥保存在安全位置后，请选择 Close (关闭)。
   - 要停用访问密钥，请选择 Make Inactive (转为非活跃)。当系统提示您确认时，请选择 Deactivate (停用）。已停用的访问密钥仍会计入您的两个访问密钥限制。
   - 要激活访问密钥，请选择 Make Active (转为活跃)。
   - 要在不再需要访问密钥时删除访问密钥，请复制访问密钥 ID，然后选择 Delete (删除)。在删除访问密钥之前，您必须选择 Deactivate (停用)。我们建议您在永久删除访问密钥之前验证该访问密钥是否已不再使用。要确认删除，请将访问密钥 ID 粘贴到文本输入字段中，然后选择 Delete (删除)。

以 IAM 用户身份登录时管理访问密钥

1. 以 IAM 用户身份登录 Amazon Web Services Management Console。有关更多信息，请参阅《IAM 用户指南》中的以 IAM 用户身份登录。
2. 在右上角的导航栏中，选择您的用户名，然后选择 My Security Credentials (我的安全凭证)。

   Tip

   如果您没有看到 My Security Credentials (我的安全凭证) 页面，您可能以联合身份用户登录，而非 IAM 用户身份登录。您可以改为创建和使用临时访问密钥 (p. 5)。
3. 请执行下列操作之一：

   - 要创建访问密钥，请选择创建访问密钥。如果您已经有两个访问密钥，则此按钮将被禁用，您必须先删除一个访问密钥，然后才能创建新的访问密钥。系统提示时，选择 Show secret access key (显示秘密访问密钥) 或者 Download .csv file (下载 .csv 文件)。这是您保存秘密访问密钥的唯一机会。将秘密访问密钥保存在安全位置后，请选择 Close (关闭)。
   - 要停用访问密钥，请选择 Make inactive (转为非活跃)。当系统提示您确认时，请选择 Deactivate (停用）。已停用的访问密钥仍会计入您的两个访问密钥限制。
   - 要激活访问密钥，请选择 Make active (转为活跃)。当系统提示进行确认时，选择 Make active (转为活跃)。
   - 要在不再需要访问密钥时删除访问密钥，请复制访问密钥 ID，然后选择 Delete (删除)。这将停用访问密钥。我们建议您在永久删除访问密钥之前验证该访问密钥是否已不再使用。要确认删除，请将访问密钥 ID 粘贴到文本输入字段中，然后选择 Delete (删除)。

经过外部身份验证的用户（身份联合验证）

您的用户可能已具有 Amazon 外部的身份，如在公司目录中。如果这些用户需要使用 Amazon 资源（或使用访问这些资源的应用程序），则这些用户也需要 Amazon 安全凭证。您可以使用 IAM 角色为身份通过您的企业或第三方身份提供程序 (IdP) 进行联合证明的用户指定权限。有关更多信息，请参阅《IAM 用户指南》中的向经过外部身份验证的用户（身份联合验证）提供访问权限。

临时访问密钥

您还可以创建和使用临时访问密钥（称为临时安全凭证）。除了访问密钥 ID 和秘密访问密钥外，临时安全凭证还包括您在使用临时安全凭证时必须发送给 Amazon 的安全令牌。临时安全凭证的优点是它们是短期使用的。在过期后，这些凭证将不再有效。在不太安全的环境中可以使用临时访问密钥，或者可以分配临时访问密钥以便向用户授予对您的 Amazon Web Services 账户 中资源的临时访问权限。例如，您可以授权来自其他 Amazon 账户的实体访问您的 Amazon Web Services 账户 中的资源（跨账户存取）。您也可以授权没

有 Amazon 安全凭证的用户访问您的 Amazon 账户中的资源（联合身份验证）。有关更多信息，请参阅`aws sts assume-role`。

您的 Amazon Web Services 账户 标识符

Amazon 将以下唯一标识符分配给每个 Amazon Web Services 账户 账户：

Amazon Web Services 账户 ID

一个 12 位数字（如 123456789012）用于唯一标识 Amazon Web Services 账户。账户 ID 不会被视为敏感信息。许多 Amazon 资源在其 Amazon Resource Name (ARN) 中包含账户 ID。账户 ID 部分将一个账户中的资源与另一个账户中的资源区分开来。如果您是 IAM 用户，您可以使用账户 ID 或账户别名登录 Amazon Web Services Management Console。

规范用户 ID

一个字母数字标识符，如 `79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be`，这是 Amazon Web Services 账户 ID 的模糊形式。规范用户 ID 不会被视为敏感信息。在使用 Amazon S3 授予对存储桶和对象的跨账户访问权限时，您可以使用此 ID 来标识 Amazon Web Services 账户。您可以按根用户或 IAM 用户的身份检索 Amazon Web Services 账户的规范用户 ID。

有关更多信息，请参阅《Amazon S3 用户指南》中的查找您的 Amazon Web Services 账户 的规范用户 ID。

您必须通过 Amazon 进行身份验证才能查看这些标识符。

Warning

不要将您的 Amazon 凭证 (p. 3)提供给需要您的 Amazon Web Services 账户 标识符与您共享 Amazon 资源的第三方。这样做将为他们提供对 Amazon Web Services 账户 的访问权限，与您拥有的访问权限相同。

查找您的 Amazon Web Services 账户 ID

您可以在 Amazon Web Services Management Console 中查找 Amazon Web Services 账户 ID。账户 ID 在控制台中的位置取决于您是以根用户还是以 IAM 用户身份登录。无论您以根用户还是以 IAM 用户身份登录，账户 ID 都是相同的。

先决条件

您必须登录 Amazon Web Services Management Console。有关更多信息，请参阅《IAM 用户指南》中的登录Amazon Web Services Management Console。

在以根用户身份登录的情况下查找您的 Amazon Web Services 账户 ID

1. 在右上角的导航栏中，选择您的账户名称或编号，然后选择 My Security Credentials (我的安全凭证)。
2. 展开 Account identifiers (账户标识符) 部分。账号显示在标签 Amazon Web Services 账户 ID 旁边。

在以 IAM 用户身份登录的情况下查找您的 Amazon Web Services 账户 ID

1. 在右上角的导航栏中，选择您的用户名，然后选择 My Security Credentials (我的安全凭证)。

   Tip

   如果您没有看到 My Security Credentials (我的安全凭证) 页面，您可能以联合身份用户登录，而非 IAM 用户身份登录。

2. 在页面顶部的 Account details（账户详细信息）下，帐号显示在标签 Amazon Web Services 账户 ID 旁边。

使用 Amazon CLI 查找您的 Amazon Web Services 账户 ID

按如下方式使用 get-caller-identity 命令：

```
aws sts get-caller-identity --query Account --output text
```

管理 Amazon 访问密钥的最佳实践

当您以编程方式使用 Amazon 时，您需要提供您的 Amazon访问密钥，以便 Amazon 可以在编程调用中验证您的身份。您的访问密钥包含访问密钥 ID（例如 `AKIAIOSFODNN7EXAMPLE`）和秘密访问密钥（例如 `wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`）。

拥有您的访问密钥的任何人将与您拥有相同的 Amazon 资源访问权限级别。因此，Amazon 全力保护您的访问密钥并确保符合我们的分担责任模型，您也应当如此。

下述步骤可以帮助您保护您的访问密钥。有关背景信息，请参阅Amazon 安全凭证 (p. 2)。

Note

贵组织的安全要求和策略可能与本主题中介绍的有所不同。此处提供的建议旨在用作一般准则。

保护或不创建您的根用户访问密钥

您必须使用访问密钥（访问密钥 ID 和秘密访问密钥）以编程方式向 Amazon 提出请求。例如，在使用 Amazon Command Line Interface、Amazon SDK 或直接 API 调用时。拥有您的 Amazon Web Services 账户 根用户的访问密钥的任何人都可以无限制地访问您 Amazon 账户中的所有资源，包括账单信息。您无法减少与您的 Amazon Web Services 账户 根用户访问密钥关联的权限。

有关更多信息，请参阅《IAM 用户指南》中的隐藏您的 Amazon Web Services 账户 根用户访问密钥。

管理 IAM 用户的访问密钥。

不是共享 Amazon Web Services 账户 根用户的凭证，而是创建单个 IAM 用户，仅授予每个用户所需的权限。有关更多信息，请参阅《IAM 用户指南》中的管理 IAM 用户的访问密钥。

使用访问密钥时，请遵守这些预防措施：

- 请勿直接将访问密钥嵌入到代码。利用 Amazon SDK 和 Amazon 命令行工具，您可以将访问密钥放在已知位置，这样就不必将其保留在代码中。

  在以下任一位置中放置访问密钥：

  - Amazon 凭证文件。Amazon 开发工具包和 Amazon CLI 自动使用您存储在 Amazon 凭证文件中的凭证。

    有关使用 Amazon 证书文件的信息，请参阅软件开发工具包文档。示例包括：《Amazon SDK for Java 开发人员指南》中的设置 Amazon 凭证和开发区域以及《Amazon Command Line Interface 用户指南》中的配置和凭证文件。

    要存储适用于 .NET 的 Amazon 开发工具包和 Amazon Tools for Windows PowerShell 的凭证，建议您使用 SDK Store。有关更多信息，请参阅《Amazon SDK for .NET 开发人员指南》中的使用 SDK 存储。
  - 环境变量. 在多租户系统上，选择用户环境变量，而不是系统环境变量。

有关使用环境变量存储凭证的更多信息，请参阅《Amazon Command Line Interface 用户指南》中的环境变量。

- 定期轮换访问密钥。定期更改访问密钥。有关详细信息，请参阅《IAM 用户指南》中的轮换访问密钥（Amazon CLI、Tools for Windows PowerShell 和 Amazon API）以及 Amazon 安全博客上的 How to Rotate Access Keys for IAM Users。
- 删除未使用的访问密钥。如果某个用户离开了贵组织，请删除相应的 IAM 用户，以使该用户无法再访问您的资源。要找出上次使用访问密钥的时间，请使用 `GetAccessKeyLastUsed` API（Amazon CLI 命令：`aws iam get-access-key-last-used`）。
- 为最敏感的操作配置多重验证。有关更多信息，请参阅《IAM 用户指南》中的在 Amazon 中使用多重身份验证（MFA）。

使用 IAM 角色而不是长期访问密钥

在许多情况下，您并不需要永不过期的长期访问密钥（如 IAM 用户访问密钥）。相反，您可以创建 IAM 角色并生成临时安全凭证。临时安全证书包括访问密钥 ID 和秘密访问密钥，以及一个指示证书何时到期的安全令牌。

在手动撤消之前，长期访问密钥将保持有效，例如与 IAM 用户和 Amazon Web Services 账户 根用户相关联的访问密钥。但是，通过 IAM 角色获取的临时安全凭证和 Amazon Security Token Service 的其他功能将在短时间内过期。凭证意外泄漏时，使用临时安全凭证可帮助降低您的风险。

在以下这些情况下使用 IAM 角色和临时安全凭证：

- 您在 Amazon EC2 实例上运行一个应用程序或 Amazon CLI 脚本。请勿直接在应用程序中使用访问密钥。请勿采取以下做法：将访问密钥传递给应用程序、将访问密钥嵌入到应用程序中、让应用程序从任何源读取密钥。相反，请定义一个对您的应用程序具有适当权限的 IAM 角色，并使用 EC2 角色启动 Amazon EC2 实例。执行此操作会将 IAM 角色与 Amazon EC2 实例相关联。这种做法还会允许应用程序获取临时安全凭证，然后再使用这些凭证对 Amazon 进行编程调用。Amazon 软件开发工具包和 Amazon CLI 可以自动获得角色的临时证书。
- 您需要授予跨账户访问权限。使用 IAM 角色建立账户之间的信任，然后向用户授予有限的账户权限来访问可信账户。有关更多信息，请参阅《IAM 用户指南》中的教程：使用 IAM 角色委派跨 Amazon 账户的访问权限。
- 您拥有一个移动应用程序。请勿将访问密钥嵌入应用程序，即使是嵌入加密存储也不允许。而应使用 Amazon Cognito 管理应用程序中的用户身份。此服务让您可以使用 Login with Amazon、Facebook、Google 或任何与 OpenID Connect（OIDC）兼容的身份提供商进行用户身份验证。然后，您可以使用 Amazon Cognito 凭证提供程序来管理应用程序用于向 Amazon 发出请求的凭证。有关更多信息，请参阅 Amazon 移动博客上的 Using the Amazon Cognito Credentials Provider。
- 您希望向 Amazon 进行联合身份验证且贵组织支持 SAML 2.0。如果您所在的组织具有支持 SAML 2.0 的身份提供程序，请将提供程序配置为使用 SAML。您可以使用 SAML 与 Amazon 交换身份验证信息，并获得一组临时安全证书。有关更多信息，请参阅《IAM 用户指南》中的关于基于 SAML 2.0 的联合身份验证.
- 您希望向 Amazon 进行联合身份验证且贵组织拥有本地身份存储。如果用户可以在组织内部进行身份验证，您可以编写一个可向他们颁发用于访问 Amazon 资源的临时安全凭证的应用程序。有关更多信息，请参阅《IAM 用户指南》中的创建允许联合身份用户访问 Amazon Web Services Management Console（自定义联合身份代理）的 URL。

使用 Amazon 访问密钥访问移动应用程序

您可以使用 Amazon 移动应用程序访问一组有限的 Amazon 服务和功能。该移动应用程序可帮助您在外出时支持事件响应。如需了解更多信息和下载应用程序，请参阅 Amazon Console Mobile Application。

您可以使用控制台密码或访问密钥登录移动应用程序。作为最佳实践，不建议使用根用户访问密钥。相反，我们强烈建议您在移动设备上除了使用密码或生物识别锁定之外，还应创建一个 IAM 用户来管理 Amazon 资

源。如果您的移动设备丢失了，您可以删除 IAM 用户的访问权限。有关为 IAM 用户生成访问密钥的更多信息，请参阅《IAM 用户指南》中的管理 IAM 用户的访问密钥。

使用访问密钥登录（移动应用程序）

1. 在移动设备上打开该应用程序。
2. 如果这是您第一次向设备添加身份，请选择 Add an identity (添加身份)，然后选择 Access keys (访问密钥)。

   如果您已使用其他身份登录，请选择菜单图标并选择 Switch identity (切换身份)。然后选择 Sign in as a different identity (以其他身份登录)，然后选择 Access keys (访问密钥)。
3. 在 Access keys (访问密钥) 页面上输入您的信息。

   - Access key ID (访问密钥 ID) – 输入您的访问密钥 ID。
   - Secret access key (秘密访问密钥) – 输入您的秘密访问密钥。
   - Identity name (身份名称) – 输入将在移动应用程序中显示的身份名称。此名称不需要与您的 IAM 用户名一致。
   - Identity PIN (身份 PIN) – 创建将来在登录时使用的个人身份识别码（PIN）。

     Note

     如果您为 Amazon 移动应用程序启用了生物识别技术，系统将提示您使用指纹或面部识别（而非 PIN）进行验证。如果生物识别失败，系统可能会提示您输入 PIN。
4. 选择 Verify and add keys (验证并添加密钥)。

   现在，您就可以使用移动应用程序访问一组选定的资源。

了解更多信息

有关确保 Amazon 账户安全的最佳实践的更多信息，请参阅以下资源：

- IAM 最佳实践 包含有关使用 Amazon Identity and Access Management（IAM）服务来帮助保护您的 Amazon 资源的建议。
- 以下页面为设置 Amazon 软件开发工具包和 Amazon CLI 以使用访问密钥提供了相关指导。
  - 有关说明，请参阅《Amazon SDK for Java 开发人员指南》中的设置用于开发的 Amazon 凭证和区域。
  - 《Amazon SDK for .NET 开发人员指南》中的使用 SDK 存储。
  - 《Amazon SDK for PHP 开发人员指南》中的为 SDK 提供凭证。
  - Boto 3（Amazon SDK for Python）文档中的配置。
  - 《Amazon Tools for Windows PowerShell 指南》中的使用 Amazon 凭证。
  - 《Amazon Command Line Interface 用户指南》中的配置和凭证文件。
- 使用 IAM 角色授予访问权限。讨论使用 .NET SDK 编写的程序在 Amazon EC2 实例上运行时如何自动获得临时安全凭证。类似信息也可用于 Amazon SDK for Java。

Amazon 安全审核指南

您应该定期审查安全配置，以确保它满足您当前的业务需求。审计可以为您提供删除不需要的 IAM 用户、角色、组和策略的机会，以确保用户和软件仅拥有必需的权限。

以下是有关系统地查看和监控 Amazon 资源的准则，以便获得安全最佳实践。

目录

何时应执行安全审核

在以下情况下，您应该审核您的安全配置：

- 定期。作为一项安全最佳实践，您应该定期执行本文档中介绍的步骤。
- 如果组织中发生变动，比如有人离职。
- 如果您停止使用一个或多个独立的 Amazon 服务。这对于删除账户中用户不再需要的权限非常重要。
- 如果您在账户中添加或删除了软件，比如 Amazon EC2 实例、Amazon OpsWorks 堆栈、Amazon CloudFormation 模板等内容上的应用程序。
- 如果您怀疑某个未授权人员可能访问了您的账户。

审核准则

查看您账户的安全配置时，请遵循这些准则：

- 全面周祥。查看安全配置的各个方面，包括您可能不经常使用的那些方面。
- 请勿假设。如果您对安全配置的某些方面（例如，特定策略背后的理由或角色存在情况）不太熟悉，请调查业务需求，直到您感到满意。
- 让事情变得简单。为了使审计（和管理）变得更简单，请使用 IAM 组、一致的命名方案和简单的策略。

审核 Amazon 账户凭证

当审核您的 Amazon 账户证书时，执行以下步骤：

1. 如果您的账户没有使用根访问密钥，您可以删除它们。我们强烈建议您创建 IAM 用户，而不是使用根访问密钥来完成 Amazon 日常工作。
2. 如果您的确需要保留账户的访问密钥，请定期轮换它们。

审核 IAM 用户

当您审计您的现有 IAM 用户时，请执行以下步骤：

1. 列出您的用户，然后删除用户（处于非活动状态）。
2. 从组中删除不必属于该组的用户。

3. 查看附加到用户所在的组的策略。请参阅有关审核 IAM 策略的提示 (p. 12)。
4. 删除用户不需要或者可能已经公开的安全证书。例如，用于应用程序的 IAM 用户无需密码（只有登录 Amazon 网站才需要密码）。同样，如果用户不使用访问密钥，则不必拥有访问密钥。有关更多信息，请参阅《IAM 用户指南》中的管理 IAM 用户的密码和管理 IAM 用户的访问密钥。

   您可以生成和下载列出您账户中所有 IAM 用户及其各个凭证状态（包括密码、访问密钥和 MFA 设备）的凭证报告。对于密码和访问密钥，凭证报告将显示多久前使用了密码或访问密钥。最近未使用的凭证可能适合做删除处理。有关更多信息，请参阅《IAM 用户指南》中的获取您的 Amazon 账户的凭证报告。
5. 定期轮换（更改）用户安全凭证，如果您已与未授权人员共享它们，请立即执行此操作。有关更多信息，请参阅《IAM 用户指南》中的管理 IAM 用户的密码和管理 IAM 用户的访问密钥。

审核 IAM 组

当您审计您的 IAM 组时，请执行以下步骤：

1. 列出您的组，然后删除组（处于未使用状态）。
2. 查看用户（位于每个组中）并删除用户（不属于这些组）。
3. 查看附加到组的策略。请参阅有关审核 IAM 策略的提示 (p. 12)。

审核 IAM 角色

当您审计您的 IAM 角色时，请执行以下步骤：

1. 列出您的角色，然后删除角色（处于未使用状态）。
2. 查看角色的信任策略。确保您知道委托人是谁，并且了解为什么账户或用户需要能够担任该角色。
3. 查看角色的访问策略，以确保其向担任该角色的人授予了合适的权限，请参阅有关审核 IAM 策略的提示 (p. 12)。

查看您的 SAML 和 OpenID Connect（OIDC）的 IAM 提供商

如果您已经创建了 IAM 实体来与 SAML 或 OIDC 身份提供商建立信任关系，请执行以下步骤：

1. 删除未使用的提供商。
2. 下载并查看每个 SAML 提供商的 Amazon 元数据文档，并确保这些文档反映了您当前的业务需求。或者，从您想与之建立信任关系的 SAML IdP 那里获取最新元数据文档，并在 IAM 中更新提供商。

审核移动应用程序

如果您已经创建了向 Amazon 提出请求的移动应用程序，请执行以下步骤：

1. 确保移动应用程序不包含嵌入式访问密钥（即使它们位于加密存储中）。
2. 通过使用为该目的设计的 API 来获取应用程序的临时证书。我们建议您使用 Amazon Cognito 来管理应用程序中的用户身份。此服务让您可以使用 Login with Amazon、Facebook、Google 或任何与 OpenID Connect（OIDC）兼容的身份提供商进行用户身份验证。然后，您可以使用 Amazon Cognito 凭证提供程序来管理应用程序用于向 Amazon 发出请求的凭证。

   如果您的移动应用程序不支持使用 Login with Amazon、Facebook、Google 或任何其他兼容 OIDC 的身份提供商进行身份验证，则可以创建代理服务器来将临时凭证分配给您的应用程序。

审核 Amazon EC2 安全配置

对每个 Amazon 区域执行以下步骤：

1. 删除未使用的或可能已经为组织之外的人员所知的 Amazon EC2 密钥对。
2. 查看 Amazon EC2 安全组：
   - 删除不再满足需求的安全组。
   - 删除不再满足需求的安全组规则。确保您知道为什么支持它们允许的端口、协议和 IP 地址范围。
3. 终止不满足业务需求，或者可能已经由组织外的人员出于未批准的目的而启动的实例。请记住，如果已通过某一角色启动实例，则在该实例上运行的应用程序可以使用该角色授予的权限来访问 Amazon 资源。
4. 取消不满足业务需求或者可能由组织外的人员提出的 Spot 实例请求。
5. 查看 Auto Scaling 组和配置。关闭任何不再满足您的需求或者可能由组织外的某个人配置的设置。

审核其他服务中的 Amazon 策略

查看使用基于资源的策略或支持其他安全机制的服务的权限。在每种情况下，确保只有具有当前业务需求的用户和角色可以访问服务资源，并且针对资源授予的权限是满足业务需求的最低要求。

- 查看 Amazon S3 存储桶策略和 ACL。
- 查看 Amazon SQS 队列策略。
- 查看 Amazon SNS 主题策略。
- 查看 Amazon OpsWorks 权限。
- 查看 Amazon KMS 密钥策略。

监控 Amazon 账户中的活动

请遵循以下监控 Amazon 活动的指导原则：

- 打开每个账户中的 Amazon CloudTrail，并在每个支持的区域中使用它。
- 定期检查 CloudTrail 日志文件。（CloudTrail 有许多合作伙伴，他们会提供用于读取和分析日志文件的工具。）
- 启用 Amazon S3 存储桶日志记录，以监控向每个存储桶提出的请求。
- 如果您认为账户遭到未授权使用，请特别注意已颁发的临时证书。如果颁发了您无法识别的临时凭证，请禁用其权限。
- 启用每个账户中的账单警报，并设置成本阈值，以便您了解费用是否超出正常使用额度。

有关审核 IAM 策略的提示

策略功能强大且非常细微，因此，学习并了解每个策略授予的权限很重要。查看策略时请使用以下准则：

- 作为最佳实践，请将策略附加到组，而不是单个用户。如果单个用户拥有策略，请确保您了解为什么该用户需要策略。
- 确保 IAM 用户、组及角色仅拥有所需的权限。
- 使用 IAM 策略模拟器对附加到用户或组的策略进行测试。
- 请记住，用户的权限是所有适用策略的结果，适用策略包括用户策略、组策略和基于资源的策略（在 Amazon S3 存储桶、Amazon SQS 队列、Amazon SNS 主题和 Amazon KMS 密钥上）。检查应用于用户的所有策略以及了解授予单个用户的一整套权限很重要。

- 请注意，通过允许用户创建 IAM 用户、组、角色或策略，并将策略附加到主要实体，可以有效地向用户授予针对账户中所有资源的权限。也就是说，可创建策略并将其附加到用户、组或角色的用户可以为自己授予任何权限。通常，不会向您不信任的用户或角色授予可以完全访问账户中资源的 IAM 权限。以下列表中包含您应仔细检查的 IAM 权限：
  - `iam:PutGroupPolicy`
  - `iam:PutRolePolicy`
  - `iam:PutUserPolicy`
  - `iam:CreatePolicy`
  - `iam:CreatePolicyVersion`
  - `iam:AttachGroupPolicy`
  - `iam:AttachRolePolicy`
  - `iam:AttachUserPolicy`
- 确保策略没有向您未使用的服务授予权限。例如，如果使用 Amazon 托管策略，请确保您账户中正在使用的 Amazon 托管策略是针对您实际使用的服务的。要找出您账户中正在使用哪些 Amazon 托管式策略，请使用 IAM `GetAccountAuthorizationDetails` API（Amazon CLI 命令：`aws iam get-account-authorization-details`）。
- 如果策略授予用户启动 Amazon EC2 实例的权限，它也可能允许 `iam:PassRole` 操作，但如果是这样，它应该明确列出允许用户传递给 Amazon EC2 实例的角色。
- 请谨慎地检查包括 `*` 的 `Action` 或 `Resource` 元素的任何值。这是仅授予用户需要的个人操作和资源的 `Allow` 访问权限的最佳实践。但是，以下是可能适合在策略中使用 `*` 的原因：
  - 策略旨在授予管理级权限。
  - 为方便起见，通配符用于一组相似的操作（例如，`Describe*`），您会因为以这种方式引用的操作的完整列表而感到轻松。
  - 通配符用于表示一类资源或一个资源路径（例如，`arn:aws:iam::account-id:users/division_abc/*`），您可以很方便地授予针对该类别或路径中所有资源的访问权限。
  - 服务操作不支持资源级权限，资源的唯一选择是 `*`。
- 检查策略名称以确保其反映了策略的功能。例如，尽管策略名称可能包括“只读”，但策略可能实际还授予了写入或更改权限。

了解更多信息

有关管理 IAM 资源的信息，请参阅以下内容：

- 《IAM 用户指南》中的 IAM 用户和组。
- 《IAM 用户指南》中的权限和策略。
- 《IAM 用户指南》中的 IAM 角色（委托和联合）。
- 《使用 IAM 策略模拟器》指南中的 IAM 策略模拟器。

有关 Amazon EC2 安全的更多信息，请参阅以下内容：

- 《适用于 Linux 实例的 Amazon EC2 用户指南》中的网络和安全性。
- Amazon 安全博客上的 Demystifying EC2 Resource-Level Permissions。

有关监控 Amazon 账户的更多信息，请参阅 re:Invent 2013 视频演示云中的入侵检测。

Amazon 资源

以下各页提供了可帮助您使用 Amazon 资源的信息。

目录

为 Amazon 资源添加标签

您可以将自己的元数据以标签的形式分配给 Amazon 资源。每个标签都是由用户定义的键和值组成的标签。标签可帮助您管理、识别、组织、搜索和筛选资源。您可以创建标签，按用途、所有者、环境或其他标准对资源进行分类。

每个标签具有两个部分：

- 标签键（例如，`CostCenter`、`Environment` 或 `Project`）。标签键区分大小写。
- 标签值（例如，`111122223333` 或 `Production`）。与标签键一样，标签值区分大小写。

您可使用标签，按用途、所有者、环境或其他标准对资源进行分类。

您可以在 Amazon 中为所有产生成本的服务标记资源。对于以下服务，Amazon 推荐更新的替代方案，它们能够更好地满足客户对于标记的使用情形。

- Amazon Cloud Directory
- Amazon CloudSearch
- Amazon Cognito Sync
- Amazon Data Pipeline
- Amazon DeepLens
- Amazon Elastic Transcoder
- Amazon Machine Learning
- Amazon OpsWorks Stacks
- Amazon S3 Glacier Direct
- Amazon SimpleDB
- Amazon WorkSpaces Application Manager (Amazon WAM)

最佳实践

在为 Amazon 资源创建标记策略时，请遵循最佳实践：

- 请勿在标签中添加个人身份信息（PII）或其他机密或敏感信息。标签可供许多 Amazon 服务访问，包括计费。标签不适合用于私有或敏感数据。

- 对标签使用标准化的区分大小写格式，并跨所有资源类型一致地应用该格式。
- 考虑支持多种用途的标签准则，如管理资源访问控制、成本跟踪、自动化和组织。
- 使用自动化工具来帮助管理资源标签。Amazon Resource Groups 和 Resource Groups Tagging API 可以对标签进行编程控制，从而能够更简单地自动管理、搜索和筛选标签与资源。
- 使用过多的标签而不是过少的标签。
- 请记住，更改标签以适应不断变化的业务需求很容易，但要考虑未来更改的后果。例如，更改访问控制标签意味着您还必须更新引用这些标签并控制对资源的访问的策略。
- 您可以通过使用 Amazon Organizations 创建和部署标签策略，自动强制执行贵组织选择采用的标记标准。标签策略使您能够指定标记规则，这些规则可以定义有效密钥名称和对每个密钥有效的值。您可以选择仅监控，从而使您有机会评估和清理现有标签。一旦您的标签符合您选择的标准，您就可以在标签策略中启用强制执行，以防止创建不合规的标签。有关更多信息，请参阅《Amazon Organizations 用户指南》中的标签策略。

为类别添加标签

最有效地使用标签的公司通常会创建与业务相关的标签分组，以便按照技术、业务和安全维度整理其资源。使用自动化流程管理其基础设施的公司还包括其他特定于自动化的标签。

技术标签	自动化标签	企业标签	安全标签
• 名称 – 标识各项资源 • 应用程序 ID – 标识与特定应用程序相关的资源 • 应用程序角色 – 描述特定资源（如 Web 服务器、消息代理、数据库）的功能 • 群集 – 标识共享通用配置并为应用程序执行特定功能的资源群 • 环境 – 区分开发资源、测试资源和生产资源 • 版本 – 帮助区分资源或应用程序的版本	• 日期/时间 – 标识应启动、停止、删除或轮换资源的日期或时间 • 选择进入/选择退出 – 指示资源是否应包含在自动化活动中，例如启动、停止或调整实例大小 • 安全性 – 确定要求，例如加密或启用 Amazon VPC 流日志；确定需要额外审查的路由表或安全组	• 项目 – 标识资源支持的项目 • 所有者 – 标识谁负责资源 • 成本中心/业务单位 – 标识与资源关联的成本中心或业务单位，通常用于成本分配和跟踪 • 客户 – 标识由特定的资源组提供服务的特定客户端	• 机密性 – 资源支持的特定数据机密性级别的标识符。 • 合规性 – 必须遵守特定合规性要求的工作负载的标识符

标签命名限制和要求

标签应遵循以下基本命名和使用要求：

- 每个资源最多可以有 50 个用户创建的标签。
- 以 `aws:` 开头的系统创建标签将保留供 Amazon 使用，并且不计入此限制。您无法编辑或删除以 `aws:` 前缀开头的标签。
- 对于每个资源，每个标签键都必须是唯一的，每个标签键只能有一个值。
- 标签键必须包含 1 到 128 个 Unicode 字符，并且以 UTF-8 格式表示。
- 标签值必须包含 0 到 256 个 Unicode 字符，并且以 UTF-8 格式表示。
- 允许的字符因 Amazon 服务而异。要了解可以使用哪些字符来标记特定 Amazon 服务中的资源，请参阅其文档。通常，允许使用的字符包括可用 UTF-8 表示的字母、数字和空格，以及以下字符：_ . : / = + - @。

- 标签键和值区分大小写。最佳实践是，决定利用标签的策略并在所有资源类型中一致地实施该策略。例如，决定是使用 `Costcenter`、`costcenter` 还是 `CostCenter`，以及是否对所有标签使用相同的约定。避免将类似的标签用于不一致的案例处理。

常见标签策略

可以使用以下标记策略帮助识别和管理 Amazon 资源。

目录

资源整理标签

标签是在 Amazon Web Services Management Console中整理 Amazon 资源的好方法 。您可以配置标签来与资源一起显示，并且可以按标签进行搜索和筛选。使用 Amazon Resource Groups 服务，您可以基于一个或多个标签或部分标签创建 Amazon 资源组。您还可以根据组在 Amazon CloudFormation 堆栈中的出现情况创建组。使用 Resource Groups 和标签编辑器，您可以在一个位置整合和查看由多个服务、资源和区域组成的应用程序的数据。

成本分配标签

Amazon Cost Explorer 和详细的账单报告可让您按标签细分 Amazon 成本。通常情况下，您使用业务标签（例如成本中心/业务部门、客户或项目）来将 Amazon 成本与传统成本分配维度关联起来。但是，成本分配报告中可以包含任何标签。这使您可以将成本与技术或安全维度（例如特定应用程序、环境或合规性项目）关联起来。以下是部分成本分配报告的示例。

Total Cost	user:Owner	user:Stack	user:Cost Center	user:Application
0.95	DbAdmin	Test	80432	Widget2
0.01	DbAdmin	Test	80432	Widget2
3.84	DbAdmin	Prod	80432	Widget2
6.00	DbAdmin	Test	78925	Widget1
234.63	SysEng	Prod	78925	Widget1
0.73	DbAdmin	Test	78925	Widget1
0.00	DbAdmin	Prod	80432	Portal
2.47	DbAdmin	Prod	78925	Portal

对于某些服务，您可以使用 Amazon 生成的 `createdBy` 标签进行成本分配，以帮助考虑如果未使用时可能未分类的资源。`createdBy` 标签仅适用于受支持的 Amazon 服务和资源。其值包含与特定 API 或控制台事件关联的数据。有关更多信息，请参阅《Amazon Billing and Cost Management 用户指南》中的 Amazon 生成的成本分配标签。

自动化标签

资源或特定于服务的标签通常用于在自动化活动期间筛选资源。自动化标签用于选择加入或退出自动化任务，或识别要存档、更新或删除的资源的特定版本。例如，您可以运行自动 `start` 或 `stop` 脚本，这些脚本可在非工作时间内关闭开发环境以降低成本。在这种情况下，Amazon Elastic Compute Cloud (Amazon

EC2) 实例标签是标识要选择退出此操作的实例的简单方法。对于查找和删除过时、过期或滚动 Amazon EBS 快照的脚本，快照标签可以添加额外的搜索条件维度。

访问控制标签

IAM 策略支持基于标签的条件，使您能够根据特定标签或标签值约束 IAM 权限。例如，IAM 用户或角色权限可以包含条件，以根据其标签将 EC2 API 调用限制到特定环境（例如开发、测试或生产）。可以使用相同的策略将 API 调用限制为特定的 Amazon Virtual Private Cloud (Amazon VPC) 网络。对基于标签的资源级 IAM 权限的支持是特定于服务的。使用基于标签的条件进行访问控制时，请务必定义和限制谁可以修改标签。有关使用标签控制 API 对 Amazon 资源的访问的更多信息，请参阅《IAM 用户指南》中的与 IAM 配合使用的 Amazon 服务。

标签监管

有效的标签策略使用标准化标签，并以编程方式在跨各种 Amazon 资源一致地应用标签。您可以使用反应式和主动式方法来管理 Amazon 环境中的标签。

- 反应式监管 用于查找未使用资源组标记 API、Amazon Config Rules 和自定义脚本等工具正确标记的资源。要手动查找资源，您可以使用标签编辑器和详细账单报告。
- 主动式监管使用 Amazon CloudFormation、Amazon Service Catalog、Amazon Organizations 中的标签策略或 IAM 资源级权限等工具，以确保在创建资源时始终应用标准化标签。

  例如，您可以使用 Amazon CloudFormation `Resource Tags` 属性将标签应用于资源类型。在 Amazon Service Catalog 中，您可以添加在产品启动时自动组合和应用于产品的产品组合和产品标签。更严格的主动监管形式包括自动化任务。例如，您可以使用资源组标记 API 搜索 Amazon 环境的标签，或者运行脚本来隔离或删除标记不当的资源。

了解更多信息

此页面提供了有关标记 Amazon 资源的一般信息。有关标记特定 Amazon 服务中的资源的更多信息，请参阅其文档。以下内容也是有关标记的有用信息来源：

- 有关 Amazon Resource Groups Tagging API 的更多信息，请参阅《资源组标记 API 参考指南》。
- 有关使用标签编辑器的信息，请参阅《Amazon Resource Groups 用户指南》中的使用标签编辑器。
- 有关使用标签控制对 Amazon 资源的访问的信息，请参阅《IAM 用户指南》中的使用 IAM 标签控制访问。

Amazon Resource Name（ARN）

Amazon Resource Name（ARN）唯一标识 Amazon 资源。当您需要在 Amazon 全局环境中（比如 IAM 策略、Amazon Relational Database Service (Amazon RDS) 标签和 API 调用中）明确指定一项资源时，我们要求使用 ARN。

服务授权参考列出了您可以在 IAM 策略中使用的 ARN。

ARN 格式

以下是 ARN 的一般格式。特定格式取决于资源。要使用 ARN，请将*##* 文本替换为特定于资源的信息。请注意，某些资源的 ARN 忽略了区域、账户 ID 或同时忽略了这两者。

```
arn:partition:service:region:account-id:resource-id
arn:partition:service:region:account-id:resource-type/resource-id
arn:partition:service:region:account-id:resource-type:resource-id
```

partition

资源所在的分区。分区 是一组 Amazon 区域。每个 Amazon 账户的作用域为一个分区。

以下是支持的分区：

- `aws` - Amazon 区域
- `aws-cn` – 中国区域
- `aws-us-gov` - Amazon GovCloud (US) 区域

service

标识 Amazon 产品的服务命名空间。例如，`s3` 代表 Amazon S3。要查找服务命名空间，请打开服务授权参考，打开该服务的页面，并在第一句中找到短语“服务前缀”。例如，以下文本显示在 Amazon S3 页面的第一句中：

```
(service prefix: s3)
```

region

区域代码。

account-id

拥有资源的 Amazon 账户的 ID（不含连字符）。例如：`123456789012`。

resource-id

资源标识符。ARN 的这一部分可以是资源的名称或 ID，也可以是资源路径 (p. 18)。例如，`user/Bob` 表示 IAM 用户，或 `instance/i-1234567890abcdef0` 表示一个 EC2 实例。某些资源标识符包括父资源 (sub-resource-type/parent-resource/sub-resource) 或限定符（例如版本）(resource-type:resource-name:qualifier)。

ARN 中的路径

资源 ARN 可以包含路径。例如，在 Amazon S3 中，资源标识符是一个对象名称，它可以包含斜杠（`/`）来形成路径。同样，IAM 用户名称和组名也可以包含路径。

路径可以包含一个通配符，即星号 (*)。例如，当您在编写 IAM 策略时，可以按以下所示使用通配符来指定包含路径 `product_1234` 的所有 IAM 用户：

```
arn:aws:iam::123456789012:user/Development/product_1234/*
```

同样，您可以指定 `user/*` 来表示所有用户，或者指定 `group/*` 来表示所有组，如以下示例所示：

```
"Resource":"arn:aws:iam::123456789012:user/*"
"Resource":"arn:aws:iam::123456789012:group/*"
```

以下示例显示了 Amazon S3 存储桶的 ARN，其中的资源名称包含一个路径：

```
arn:aws:s3:::my_corporate_bucket/*
arn:aws:s3:::my_corporate_bucket/Development/*
```

不正确的通配符使用

您不能在 ARN 指定资源类型的部分使用通配符，比如 IAM ARN 中的 `user` 一词。例如，不允许执行以下操作。

```
arn:aws:iam::123456789012:u*   <== not allowed
```

Amazon IP 地址范围

Amazon Web Services (Amazon) 以 JSON 格式发布其当前的 IP 地址范围。要查看当前范围，请下载 `.json` 文件。要维护历史记录，请将连续版本的 `.json` 文件保存在系统上。要确定自上次保存文件以来是否发生更改，请检查当前文件中的发布时间，并将其与上次保存文件中的发布时间进行比较。

通过自带 IP 地址（BYOIP）引入到 Amazon 的 IP 地址范围不包含在 `.json` 文件内。

目录

下载

下载 ip-ranges.json。

如果您以编程方式访问此文件，您有责任确保仅在成功验证服务器提供的 TLS 证书之后，应用程序才能下载文件。

语法

`ip-ranges.json` 的语法如下。

```
{
  "syncToken": "0123456789",
  "createDate": "yyyy-mm-dd-hh-mm-ss",
  "prefixes": [
    {
      "ip_prefix": "cidr",
      "region": "region",
      "network_border_group": "network_border_group",
      "service": "subset"
    }
  ],
  "ipv6_prefixes": [
    {
      "ipv6_prefix": "cidr",
      "region": "region",
      "network_border_group": "network_border_group",
      "service": "subset"
    }
  ]
}
```

syncToken

采用 Unix 纪元时间格式的发布时间。

类型：字符串

示例：`"syncToken": "1416435608"`

createDate

发布日期和时间，采用 UTC YY-MM-DD-hh-mm-ss 格式。

类型：字符串

示例：`"createDate": "2014-11-19-23-29-02"`

prefixes

IPv4 地址范围的 IP 前缀。

类型：数组

ipv6_prefixes

IPv6 地址范围的 IP 前缀。

类型：数组

ip_prefix

用 CIDR 表示法指定的公有 IPv4 地址范围。请注意，Amazon 可在更具体的范围内公布前缀。例如，文件中的前缀 96.127.0.0/17 可公布为 96.127.0.0/21、96.127.8.0/21、96.127.32.0/19 和 96.127.64.0/18。

类型：字符串

示例：`"ip_prefix": "198.51.100.2/24"`

ipv6_prefix

用 CIDR 表示法指定的公有 IPv6 地址范围。请注意，Amazon 可在更具体的范围内公布前缀。

类型：字符串

示例：`"ipv6_prefix": "2001:db8:1234::/64"`

network_border_group

网络边界组的名称，这是 Amazon 通告 IP 地址的可用区或 Local Zones 的唯一集合。

类型：字符串

示例：`"network_border_group": "us-west-2-lax-1"`

区域

Amazon 区域或 `GLOBAL`（针对边缘站点）。`CLOUDFRONT` 和 `ROUTE53` 范围是 `GLOBAL`。

类型：字符串

有效值：`ap-east-1` | `ap-northeast-1` | `ap-northeast-2` | `ap-northeast-3` | `ap-south-1` | `ap-southeast-1` | `ap-southeast-2` | `ca-central-1` | `cn-north-1` | `cn-northwest-1` | `eu-central-1` | `eu-north-1` | `eu-west-1` | `eu-west-2` | `eu-west-3` | `me-central-1` | `me-south-1` | `sa-east-1` | `us-east-1` | `us-east-2` | `us-gov-east-1` | `us-gov-west-1` | `us-west-1` | `us-west-2` | `GLOBAL`

示例：`"region": "us-east-1"`

service

IP 地址范围的子集。为 `API_GATEWAY` 列出的地址仅为出口 IP 地址。指定 `AMAZON` 可获得所有 IP 地址范围（这意味着每个子集也在 `AMAZON` 子集中）。但是，某些 IP 地址范围仅在 `AMAZON` 子集中（这意味着它们不会再包含在其他子集中）。

类型：字符串

有效值：AMAZON | AMAZON_APPFLOW | AMAZON_CONNECT | API_GATEWAY | CHIME_MEETINGS | CHIME_VOICECONNECTOR | CLOUD9 | CLOUDFRONT | CLOUDFRONT_ORIGIN_FACING | CODEBUILD | DYNAMODB | EBS | EC2 | EC2_INSTANCE_CONNECT | GLOBALACCELERATOR | KINESIS_VIDEO_STREAMS | ROUTE53 | ROUTE53_HEALTHCHECKS | ROUTE53_HEALTHCHECKS_PUBLISHING | ROUTE53_RESOLVER | S3 | WORKSPACES_GATEWAYS

示例："service": "AMAZON"

筛选 JSON 文件

您可以下载命令行工具以帮助您筛选出自己所要查找的信息。

Windows

Amazon Tools for Windows PowerShell 包含 cmdlet `Get-AWSPublicIpAddressRange` 以便分析此 JSON 文件。以下示例展示了其用法。有关更多信息，请参阅查询 Amazon 的公有 IP 地址范围和 Get-AWSPublicIpAddressRange。

Example 1. 获取创建日期

```
PS C:\> Get-AWSPublicIpAddressRange -OutputPublicationDate

Wednesday, August 22, 2018 9:22:35 PM
```

Example 2. 获取特定区域的信息

```
PS C:\> Get-AWSPublicIpAddressRange -Region us-east-1

IpPrefix        Region      NetworkBorderGroup         Service
--------        ------       -------                   -------
23.20.0.0/14    us-east-1    us-east-1                 AMAZON
50.16.0.0/15    us-east-1    us-east-1                 AMAZON
50.19.0.0/16    us-east-1    us-east-1                 AMAZON
...
```

Example 3. 获取所有 IP 地址

```
PS C:\> (Get-AWSPublicIpAddressRange).IpPrefix
23.20.0.0/14
27.0.0.0/22
43.250.192.0/24
...
2406:da00:ff00::/64
2600:1fff:6000::/40
2a01:578:3::/64
2600:9000::/28
```

Example 4. 获取所有 IPv4 地址

```
PS C:\> Get-AWSPublicIpAddressRange | where {$_.IpAddressFormat -eq "Ipv4"} | select
 IpPrefix

IpPrefix
--------
```

```
23.20.0.0/14
27.0.0.0/22
43.250.192.0/24
...
```

Example 5. 获取所有 IPv6 地址

```
PS C:\> Get-AWSPublicIpAddressRange | where {$_.IpAddressFormat -eq "Ipv6"} | select
 IpPrefix

IpPrefix
--------
2a05:d07c:2000::/40
2a05:d000:8000::/40
2406:dafe:2000::/40
...
```

Example 6. 获取特定服务的所有 IP 地址

```
PS C:\> Get-AWSPublicIpAddressRange -ServiceKey CODEBUILD | select IpPrefix

IpPrefix
--------
52.47.73.72/29
13.55.255.216/29
52.15.247.208/29
...
```

Linux

以下示例命令使用 jq 工具分析 JSON 文件的本地副本。

Example 1. 获取创建日期

```
$ jq .createDate < ip-ranges.json

"2016-02-18-17-22-15"
```

Example 2. 获取特定区域的信息

```
$ jq '.prefixes[] | select(.region=="us-east-1")' < ip-ranges.json

{
  "ip_prefix": "23.20.0.0/14",
  "region": "us-east-1",
  "network_border_group": "us-east-1",
  "service": "AMAZON"
},
{
  "ip_prefix": "50.16.0.0/15",
  "region": "us-east-1",
  "network_border_group": "us-east-1",
  "service": "AMAZON"
},
{
  "ip_prefix": "50.19.0.0/16",
  "region": "us-east-1",
  "network_border_group": "us-east-1",
  "service": "AMAZON"
```

```
},
...
```

Example 3. 获取所有 IPv4 地址

```
$ jq -r '.prefixes | .[].ip_prefix' < ip-ranges.json

23.20.0.0/14
27.0.0.0/22
43.250.192.0/24
...
```

Example 4. 获取所有 IPv6 地址

```
$ jq -r '.ipv6_prefixes | .[].ipv6_prefix' < ip-ranges.json

2a05:d07c:2000::/40
2a05:d000:8000::/40
2406:dafe:2000::/40
...
```

Example 5. 获取特定服务的所有 IPv4 地址

```
$ jq -r '.prefixes[] | select(.service=="CODEBUILD") | .ip_prefix' < ip-ranges.json

52.47.73.72/29
13.55.255.216/29
52.15.247.208/29
...
```

Example 6. 获取特定区域中的特定服务的所有 IPv4 地址

```
$ jq -r '.prefixes[] | select(.region=="us-east-1") | select(.service=="CODEBUILD")
 | .ip_prefix' < ip-ranges.json

34.228.4.208/28
```

Example 7. 获取特定网络边界组的信息

```
$ jq -r '.prefixes[] | select(.region=="us-west-2") | select(.network_border_group=="us-
west-2-lax-1") | .ip_prefix' < ip-ranges.json
70.224.192.0/18
52.95.230.0/24
15.253.0.0/16
...
```

实施出口控制

要仅允许实例访问 Amazon 服务，可使用规则创建安全组，这些规则允许出站流量流向 AMAZON 列表中的 CIDR 块（排除也在 EC2 列表中的 CIDR 块）。EC2 列表中的 IP 地址可以分配给 EC2 实例。

Windows PowerShell

以下 PowerShell 示例显示了如何获取位于 AMAZON 列表中但不位于 EC2 列表中的 IP 地址。复制脚本并将其保存到名为 Select_address.ps1 的文件中。

```
$amazon_addresses = Get-AWSPublicIpAddressRange -ServiceKey amazon
$ec2_addresses = Get-AWSPublicIpAddressRange -ServiceKey ec2

ForEach ($address in $amazon_addresses)
{
    if( $ec2_addresses.IpPrefix -notcontains $address.IpPrefix)
    {
       ($address).IpPrefix
    }
}
```

您可以按如下方式运行此脚本：

```
PS C:\> .\Select_address.ps1
13.32.0.0/15
13.35.0.0/16
13.248.0.0/20
13.248.16.0/21
13.248.24.0/22
13.248.28.0/22
27.0.0.0/22
43.250.192.0/24
43.250.193.0/24
...
```

jq

以下示例说明如何获取位于 AMAZON 列表中但不位于 EC2 列表中的 IP 地址（针对所有区域）：

```
jq -r '[.prefixes[] | select(.service=="AMAZON").ip_prefix] - [.prefixes[] |
 select(.service=="EC2").ip_prefix] | .[]' < ip-ranges.json

52.94.22.0/24
52.94.17.0/24
52.95.154.0/23
52.95.212.0/22
54.239.0.240/28
54.239.54.0/23
52.119.224.0/21
...
```

以下示例说明如何将结果筛选到一个区域：

```
jq -r '[.prefixes[] | select(.region=="us-east-1" and .service=="AMAZON").ip_prefix] -
 [.prefixes[] | select(.region=="us-east-1" and .service=="EC2").ip_prefix] | .[]' < ip-
ranges.json
```

Python

以下 python 脚本显示了如何获取位于 AMAZON 列表中但不位于 EC2 列表中的 IP 地址。复制脚本并将其保存到名为 get_ips.py 的文件中。

```
#!/usr/bin/env python
import requests

ip_ranges = requests.get('https://ip-ranges.amazonaws.com/ip-ranges.json').json()
['prefixes']
amazon_ips = [item['ip_prefix'] for item in ip_ranges if item["service"] == "AMAZON"]
```

```
ec2_ips = [item['ip_prefix'] for item in ip_ranges if item["service"] == "EC2"]

amazon_ips_less_ec2=[]

for ip in amazon_ips:
    if ip not in ec2_ips:
        amazon_ips_less_ec2.append(ip)

for ip in amazon_ips_less_ec2: print(str(ip))
```

您可以按如下方式运行此脚本：

```
$ python ./get_ips.py
13.32.0.0/15
13.35.0.0/16
13.248.0.0/20
13.248.16.0/21
13.248.24.0/22
13.248.28.0/22
27.0.0.0/22
43.250.192.0/24
43.250.193.0/24
...
```

发布说明

下表介绍对Amazon IP 地址范围的更新。我们还会在推出每个区域时添加新的区域代码。

描述	发行日期
添加了 CLOUDFRONT_ORIGIN_FACING 服务代码。	2021 年 10 月 12 日
添加了 ROUTE53_RESOLVER 服务代码。	2021 年 6 月 24 日
添加了 EBS 服务代码。	2021 年 5 月 12 日
添加了 KINESIS_VIDEO_STREAMS 服务代码。	2020 年 11 月 19 日
添加了 CHIME_MEETINGS 和 CHIME_VOICECONNECTOR 服务代码。	2020 年 6 月 19 日
添加了 AMAZON_APPFLOW 服务代码。	2020 年 6 月 9 日
增加了对网络边界组的支持。	2020 年 4 月 7 日
添加了 WORKSPACES_GATEWAYS 服务代码。	2020 年 3 月 30 日
添加了 ROUTE53_HEALTHCHECK_PUBLISHING 服务代码。	2020 年 1 月 30 日
添加了 API_GATEWAY 服务代码。	2019 年 9 月 26 日
添加了 EC2_INSTANCE_CONNECT 服务代码。	2019 年 6 月 26 日
添加了 DYNAMODB 服务代码。	2019 年 4 月 25 日
添加了 GLOBALACCELERATOR 服务代码。	2018 年 12 月 20 日

描述	发行日期
添加了 AMAZON_CONNECT 服务代码。	2018 年 6 月 20 日
添加了 CLOUD9 服务代码。	2018 年 6 月 20 日
添加了 CODEBUILD 服务代码。	2018 年 4 月 19 日
添加了 S3 服务代码。	2017 年 2 月 28 日
添加了对 IPv6 地址范围的支持。	2016 年 8 月 22 日
首次发布	2014 年 11 月 19 日

了解更多信息

- AMAZON_APPFLOW – IP 地址范围
- AMAZON_CONNECT – 设置您的网络
- CLOUDFRONT – CloudFront 边缘服务器的位置和 IP 地址范围
- DYNAMODB – IP 地址范围
- EC2 – 公有 IPV4 地址
- EC2_INSTANCE_CONNECT – 设置 EC2 Instance Connect
- GLOBALACCELERATOR – Global Accelerator 边缘服务器的位置和 IP 地址范围
- ROUTE53 – Amazon Route 53 服务器的 IP 地址范围
- ROUTE53_HEALTHCHECKS – Amazon Route 53 服务器的 IP 地址范围
- ROUTE53_HEALTHCHECKS_PUBLISHING – Amazon Route 53 服务器的 IP 地址范围
- WORKSPACES_GATEWAYS–PCoIP 网关服务器

Amazon API

以下各页提供了在使用 Amazon API 时非常有用的信息。

目录

Amazon 中的错误重试和指数退避

网络上的大量组件 (例如 DNS 服务器、交换机、负载均衡器等) 都可能在某个指定请求生命周期中的任一环节出现问题。在联网环境中，处理这些错误响应的常规技术是在客户应用程序中实施重试。此技术可以提高应用程序的可靠性和降低开发人员的操作成本。

每个 Amazon 开发工具包都会实现自动重试逻辑。Amazon SDK for Java会自动重试请求，您可以使用 `ClientConfiguration` 类配置重试设置。例如，对于发出的请求采用最低延迟并且不想重试的网页，您可能会希望关闭重试逻辑。您可以使用 `ClientConfiguration` 类，并且为 `maxErrorRetry` 提供 `0` 这个值，从而设置为不重试。

如果您没有使用 Amazon 开发工具包，则应当对收到服务器错误 (5xx) 或限制错误的原始请求执行重试。但是，客户端错误 (4xx) 表示您需要对请求本身进行修改，纠正该问题，然后再重试。

除了简单重试之外，每个 Amazon SDK 还实施指数回退算法来实现更好的流程控制。指数退避的原理是对于连续错误响应，重试等待间隔越来越长。您应该实施最长延迟间隔和最大重试次数。最长延迟间隔和最大重试次数不一定是固定值，并且应当根据正在执行的操作和其他本地因素（例如网络延迟）进行设置。

大多数指数回退算法会利用抖动（随机延迟）防止连续的冲突。由于在这些情况下您并未尝试避免此类冲突，因此无需使用此随机数字。但是，如果使用并发客户端，抖动可帮助您更快地成功执行请求。有关更多信息，请参阅指数退避和抖动的博文。

以下伪代码显示了一种使用增量延迟轮询状态的方法。

```
Do some asynchronous operation.

retries = 0

DO
    wait for (2^retries * 100) milliseconds

    status = Get the result of the asynchronous operation.

    IF status = SUCCESS
        retry = false
    ELSE IF status = NOT_READY
        retry = true
    ELSE IF status = THROTTLED
        retry = true
    ELSE
        Some other error occurred, so stop calling the API.
        retry = false
    END IF
```

```
    retries = retries + 1

WHILE (retry AND (retries < MAX_RETRIES))
```

签署 Amazon API 请求

Important

Amazon SDK 、Amazon Command Line Interface (Amazon CLI) 和其他 Amazon 工具会使用您在配置工具时指定的访问密钥为您签署 API 请求。当您使用这些工具时，您不必了解如何签署这些 API 请求。以下文档说明了如何签署 API 请求，但仅适用于您编写自己的代码来发送和签署 Amazon API 请求的情况。建议使用 Amazon SDK 或其他 Amazon 工具来发送 API 请求，而不是编写自己的代码。

当您将 API 请求发送到Amazon时，您需要签署请求，以便Amazon能够识别发送它们的用户。您将使用您的 Amazon 访问密钥来签署请求，该访问密钥包含访问密钥 ID 和秘密访问密钥。有些请求不需要签署，包括对 Amazon Simple Storage Service (Amazon S3) 的匿名请求以及 Amazon Security Token Service (Amazon STS) 中的部分 API 操作 (例如 `AssumeRoleWithWebIdentity`)

何时签署请求

当您编写自定义代码来将 API 请求发送到Amazon时，就需要包含用于签署请求的代码。您可能出于以下原因来执行该操作：

- 您正在使用的编程语言没有对应的 Amazon 开发工具包。
- 您希望完全控制将请求发送到 Amazon 的方式。

在使用 Amazon CLI 或其中一个 Amazon SDK 时，您无需签署请求。这些工具会为您计算签名，并管理连接详细信息、处理请求重试和提供错误处理。在大多数情况下，它们还包含示例代码、教程和其他资源，可帮助您开始编写与 Amazon 交互的应用程序。

为什么签署请求

签名过程通过以下方式帮助保护请求：

- 验证请求者的身份

  签名可以确保请求是由某个具有有效访问密钥的用户发送的。有关更多信息，请参阅了解并获取您的 Amazon 凭证 (p. 3)。
- 保护传输中的数据

  为了防止传输时请求被篡改，一些请求元素将用于计算请求的哈希 (摘要) ，得到的哈希值将包括在请求中。在 Amazon 服务收到请求时，它将使用相同信息计算哈希，并将其与您的请求中包括的哈希值进行匹配。如果值不匹配，Amazon 将拒绝请求。
- 防止潜在的反演攻击

  在大多数情况下，请求必须在请求中的时间戳的 5 分钟内到达 Amazon。否则，Amazon 将拒绝该请求。

签署请求

要对请求签名，请先计算请求的哈希 (摘要)。然后，您使用哈希值、来自请求的其他一些信息以及您的私密访问密钥，计算另一个称为签名 的哈希。接下来，您可以通过以下方式之一将签名添加到请求：

- 使用 HTTP `Authorization` 标头。
- 将查询字符串值添加到请求中。在本例中，由于签名是 URL 的一部分，因此这类 URL 被称为预签名 URL。

签名版本

Amazon 支持 Signature Version 4（SigV4）和 Signature Version 2（SigV2）。所有 Amazon Web Services 区域 中的全部 Amazon 服务都支持 SigV4，但 Amazon SimpleDB 除外，它要求使用 SigV2。Amazon SDK（包括 Amazon CLI）会自动将 SigV4 应用于支持它的所有服务。如果手动签署 API 请求，您需执行同样的操作。

Amazon 正在推出 SigV4 的扩展，称为签名版本 4A（SigV4A）。此扩展支持签名在多个 Amazon Web Services 区域 中有效。这是签署多区域 API 请求所必需的，例如 Amazon S3 多区域访问点。Amazon SDK 和 Amazon CLI 支持 SigV4A，并在需要时自动使用。

> **Note**
>
> 要将 SigV4A 与临时安全凭证一起使用（例如，在使用 IAM 角色时），请确保从 Amazon Security Token Service (Amazon STS) 中的区域端点请求临时凭证。不要使用 Amazon STS (`sts.amazonaws.com`) 的全局端点，因为默认情况下，全局端点的临时凭证不适用于 SigV4A。您可以使用任何 Amazon STS 的区域端点。

Signature Version 4 签名流程

> **Important**
>
> Amazon SDK、Amazon Command Line Interface (Amazon CLI) 和其他 Amazon 工具会使用您在配置工具时指定的访问密钥为您签署 API 请求。当您使用这些工具时，您不必了解如何签署这些 API 请求。以下文档说明了如何签署 API 请求，但仅适用于您编写自己的代码来发送和签署 Amazon API 请求的情况。建议使用 Amazon SDK 或其他 Amazon 工具来发送 API 请求，而不是编写自己的代码。

Signature Version 4（SigV4）是将身份验证信息添加到通过 HTTP 发送的 Amazon API 请求的流程。为了安全起见，大多数发给 Amazon 的请求都必须使用访问密钥签名。访问密钥包括访问密钥 ID 和秘密访问密钥，这两个访问密钥通常称为您的安全凭证。有关如何获取您账户的凭证的详细信息，请参阅了解并获取您的 Amazon 凭证 (p. 3)。

Signature Version 4 的工作原理

1. 创建规范请求。
2. 使用规范请求和其他元数据来创建供签署的字符串。
3. 从您的 Amazon 秘密访问密钥派生签名密钥。然后将该签名密钥与在上一步中创建的字符串结合使用来创建签名。
4. 将生成的签名添加到 HTTP 请求的标头中或者添加为查询字符串参数。

Amazon 服务收到请求后，将执行您完成的相同步骤来计算请求中发送的签名。之后，Amazon 会将计算得到的签名与您在请求中发送的签名进行比较。如果签名匹配，则处理请求。如果签名不匹配，则拒绝请求。

有关更多信息，请参阅以下 资源：

- 若要开始签名过程，请参阅利用 Signature Version 4 对Amazon请求进行签名 (p. 32)。
- 有关已签名请求的示例，请参阅Signature Version 4 完整签名过程的示例（Python） (p. 46)。
- 如果您有关于 Signature Version 4 的问题，请在 Amazon Identity and Access Management 论坛中发布您的问题。

Amazon Signature Version 4 请求的元素

每个使用版本 4 签名的 HTTP/HTTPS 请求都必须包含这些元素。

- 终端节点规范
- 操作
- 必需和可选参数
- 日期
- 身份验证参数

终端节点规范

这指定为 HTTP/1.1 请求中的 `Host` 标头。此标头指定您要向其发送请求的计算机的 DNS 名称，例如 `dynamodb.us-east-1.amazonaws.com`。

您必须随 HTTP/1.1 请求包含 `Host` 标头。对于 HTTP/2 请求，您可以使用 `:authority` 标头或 `Host` 标头。仅使用 `:authority` 标头以符合 HTTP/2 规范。并非所有服务都支持 HTTP/2 请求，因此请检查服务文档以了解详细信息。

终端节点通常包含服务名称和区域，您必须将这两者用作 `Credential` 身份验证参数的一部分。例如，eu-west-1 区域的 Amazon DynamoDB 终端节点为 `dynamodb.eu-west-1.amazonaws.com`。如果您未指定区域，Web 服务将使用默认区域 `us-east-1`。如果您使用的是使用全球唯一终端节点的服务（如 IAM），请使用默认区域（`us-east-1`）作为 `Credential` 身份验证参数的一部分（如本主题后面所述）。

有关 Amazon 支持的终端节点的完整列表，请参阅区域和终端节点。

操作

此元素指定您希望 Web 服务执行的操作，如 DynamoDB `CreateTable` 操作或 Amazon EC2 `DescribeInstances` 操作。指定的操作确定在请求中使用的参数。对于查询 API，此操作是一个 API 名称。对于非查询 API（如 RESTful API），请参阅服务文档以了解相应的操作。

必需和可选参数

此元素指定请求操作的参数。Web 服务中的每个操作都具有一组用于定义 API 调用的必需参数和可选参数。API 版本通常是必需参数。有关必需参数和可选参数的详细信息，请参阅服务文档。

日期

这是您发出请求的日期和时间。在请求中包括日期有助于防止第三方拦截您的请求并稍后重新提交。该日期使用 ISO8601 Basic 格式通过 `x-amz-date` 标头以 `YYYYMMDD'T'HHMMSS'Z'` 格式指定。

身份验证参数

您发送的每个请求都必须包含以下一组参数，Amazon使用这些参数来确保请求的有效性和真实性。

- 算法。您用作签名过程一部分的哈希算法。例如，如果您使用 SHA-256 创建哈希，请使用值 `AWS4-HMAC-SHA256`。
- 凭证范围。以斜杠（“/”）分隔的字符串，它通过将您的访问密钥 ID 和凭证范围组件串联起来而形成。凭证范围包括采用 YYYYMMDD 格式的日期、Amazon 区域、服务名称和特殊的终止字符串 (`aws4_request`)。例如，以下字符串表示 us-east-1 cn-north-1 区域中 IAM 请求的 `Credential` 参数。

  ```
  AKIAIOSFODNN7EXAMPLE/20111015/us-east-1/iam/aws4_request

  AKIAIOSFODNN7EXAMPLE/20111015/cn-north-1/iam/aws4_request
  ```

Important

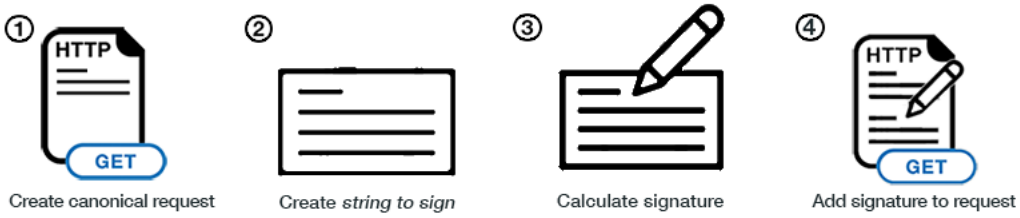
您必须为区域、服务名称和特殊的终止字符串使用小写字符。

- SignedHeaders 要在签名中包含的 HTTP/HTTPS 标头的以分号（“;”）分隔的列表。
- 签名 一个十六进制编码字符串，表示在任务 3：为 Amazon Signature Version 4 计算签名 (p. 40)中介绍的签名操作的输出。您必须使用您在 `Algorithm` 参数中指定的算法来计算签名。

若要查看示例已签名请求，请参阅Signature Version 4 完整签名过程的示例（Python） (p. 46)。

利用 Signature Version 4 对Amazon请求进行签名

本节介绍如何创建签名并将其添加发给Amazon的 HTTP 请求中。



签名步骤摘要

要创建已签名的请求，请完成以下操作：

- 任务 1：针对 Signature Version 4 创建规范请求 (p. 34)

  将请求的内容（主机、操作、标头等）组织为标准（规范）格式。规范请求是用于创建待签字符串的输入之一。
- 任务 2：创建 Signature Version 4 的待签字符串 (p. 39)

  使用规范请求和额外信息（例如算法、请求日期、凭证范围和规范请求的摘要（哈希））创建待签字符串。
- 任务 3：为 Amazon Signature Version 4 计算签名 (p. 40)

  使用 Amazon 秘密访问密钥作为初始哈希操作的密钥，对请求日期、区域和服务执行一系列加密哈希操作（HMAC 操作），从而派生签名密钥。在派生签名密钥后，通过对待签字符串执行加密哈希操作来计算签名。使用派生的签名密钥作为此操作的哈希密钥。
- 任务 4：将签名添加到 HTTP 请求 (p. 41)

  在计算签名后，将其添加到请求的 HTTP 标头或查询字符串中。

Important

Amazon SDK 可以为您处理签名计算过程，因此您无需手动完成签名过程。有关更多信息，请参阅用于 Amazon Web Services 的工具。

其他资源

以下资源全面介绍了签名过程的各个方面：

- 说明如何为 Signature Version 4 派生签名密钥的示例 (p. 43)。此页演示如何使用 Java、C#、Python、Ruby 和 JavaScript 派生签名密钥。

- Signature Version 4 完整签名过程的示例（Python）(p. 46)。这组用 Python 编写的程序提供了完整的签名过程示例。这些示例演示了如何利用 POST 请求、利用在请求标头中具有签名信息的 GET 请求以及利用在查询字符串中具有签名信息的 GET 请求进行签名。

请求中的签署过程

以下示例演示了 HTTPS 请求（不带任何签名信息）从您的客户端发送到Amazon时可能的样子。

```
GET https://iam.cn-north-1.amazonaws.com.cn/?Action=ListUsers&Version=2010-05-08 HTTP/1.1
Content-Type: application/x-www-form-urlencoded; charset=utf-8
Host: iam.cn-north-1.amazonaws.com.cn
X-Amz-Date: 20150830T123600Z
```

在完成签名任务之后，请将身份验证信息添加到请求中。您可通过两种方式添加身份验证信息：

Authorization 标头

您可使用 Authorization 标头将身份验证信息添加到请求中。尽管该 HTTP 标头名为 Authorization，但签名信息实际上是用于身份验证的，目的是确定请求方。

Authorization 标头包含以下信息：

- 用于签名的算法 (AWS4-HMAC-SHA256)
- 凭证范围（包含您的访问密钥 ID）
- 已签名标头的列表
- 计算签名。该签名基于您的请求信息，由您使用 Amazon 秘密访问密钥生成。该签名用于向 Amazon 确认您的身份。

下面的示例说明在您创建签名信息并将它添加到请求的 Authorization 标头之后，前面的请求看起来是什么样。

请注意，在实际请求中，Authorization 标头显示为一行连续的文本。为便于阅读，下面的版本已经过格式编排。

```
POST https://iam.cn-north-1.amazonaws.com.cn/?Action=ListUsers&Version=2010-05-08 HTTP/1.1
Authorization: AWS4-HMAC-SHA256
 Credential=AKIDEXAMPLE/20150830/cn-north-1/iam/aws4_request,
 SignedHeaders=content-type;host;x-amz-date,
 Signature=d37af66cc90dc26bb2e27d2a97316b729b82589b5e4648f1ae34cb83a3f546cd
content-type: application/x-www-form-urlencoded; charset=utf-8
host: iam.cn-north-1.amazonaws.com.cn
x-amz-date: 20150830T123600Z
```

查询字符串

作为一种将身份验证信息添加到 HTTP 请求标头中的替代方法，您可以将它包含在查询字符串中。查询字符串包含请求的所有部分，包括操作的名称和参数、日期以及身份验证信息。

以下示例为您演示如何通过查询字符串中的操作和身份验证信息来构造 GET 请求。

（在实际请求中，查询字符串会显示为一行连续的文本。为便于阅读，下面的版本已经用换行符编排过格式。）

```
GET https://iam.cn-north-1.amazonaws.com.cn?Action=ListUsers&Version=2010-05-08
```

```
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIDEXAMPLE%2F20150830%2Fcn-north-1%2Fiam%2Faws4_request
&X-Amz-Date=20150830T123600Z
&X-Amz-Expires=60
&X-Amz-SignedHeaders=content-type%3Bhost
&X-Amz-Signature=bbb7890b2172f0cccc6d1d5cded4e690f3e1dac299599547f3d1ceb50567e83d HTTP/1.1
content-type: application/x-www-form-urlencoded; charset=utf-8
host: iam.cn-north-1.amazonaws.com.cn
```

查询 API 中的 GET 和 POST 请求

许多Amazon服务支持的查询 API 可用来通过 HTTP `GET` 或 `POST` 发送请求。（在查询 API 中，即使您提交更改状态的请求，即查询 API 本身并非 RESTful，您也可以使用 `GET`。）因为 `GET` 请求通过查询字符串传递参数，所以它们的长度限制为 URL 的最大长度。如果请求包含大型有效负载（例如，您可能以 JSON 格式为 DynamoDB 请求上传大型 IAM policy 或发送大量参数），则通常使用 `POST` 请求。

两种类型的请求的签名过程相同。

任务 1：针对 Signature Version 4 创建规范请求

要开始签名过程，请创建一个字符串，其中包含来自标准（规范）格式的请求的信息。这可确保 Amazon 在收到请求时计算出的签名与您计算出的签名相同。

按照此处的步骤创建请求的规范版本。否则，您的版本与Amazon计算得到的版本将不匹配，请求将被拒绝。

以下示例演示了创建规范请求的伪代码。

Example 规范请求伪代码

```
CanonicalRequest =
  HTTPRequestMethod + '\n' +
  CanonicalURI + '\n' +
  CanonicalQueryString + '\n' +
  CanonicalHeaders + '\n' +
  SignedHeaders + '\n' +
  HexEncode(Hash(RequestPayload))
```

在此伪代码中，`Hash` 表示生成消息摘要的函数，通常是 SHA-256。（在该过程稍后的阶段中，您将指定要使用的哈希算法。）`HexEncode` 表示以小写字母形式返回摘要的 base-16 编码的函数。例如，`HexEncode("m")` 返回值 `6d` 而不是 `6D`。输入的每一个字节都必须表示为两个十六进制字符。

签名版本 4 不需要您使用特定字符编码来对规范请求进行编码。不过，一些 Amazon 服务可能需要特定编码。有关更多信息，请参阅该服务的文档。

以下示例演示如何构造规范形式的 IAM 请求。原始请求在从客户端发送到Amazon时可能看上去与此类似，不过此示例还不包括签名信息。

Example 请求

```
GET https://iam.cn-north-1.amazonaws.com.cn/?Action=ListUsers&Version=2010-05-08 HTTP/1.1
Host: iam.cn-north-1.amazonaws.com.cn/
Content-Type: application/x-www-form-urlencoded; charset=utf-8
X-Amz-Date: 20150830T123600Z
```

前面的示例请求是一个 GET 请求（方法），它向 Amazon Identity and Access Management（主机）发出 `ListUsers` API（操作）调用。此操作采用 `Version` 参数。

要创建规范请求，请将以下来自每个步骤的部分连接为一个字符串：

1. 首先是 HTTP 请求方法（`GET`、`PUT`、`POST` 等），后跟换行符。

   Example 请求方法

   ```
   GET
   ```

2. 添加规范 URI 参数，后跟换行符。规范 URI 是 URI 的绝对路径部分的 URI 编码版本，该版本是 URI 中的一切 - 从 HTTP 主机到开始查询字符串参数（如果有）的问号字符（“?”）。

   根据 RFC 3986 标准化 URI 路径。移除冗余和相对路径部分。路径中每个部分都必须经 URI 编码两次（Amazon S3 除外，其仅 URI 编码一次）。

   Example 使用编码的规范 URI

   ```
   /documents%2520and%2520settings/
   ```

   > **Note**
   >
   > 例外情况是，您没有使用规范的 URI 路径来提出 Amazon S3 请求。例如，如果您拥有包含名为 `my-object//example//photo.user` 的对象的存储桶，请使用该路径。如果将该路径标准化为 `my-object/example/photo.user`，则会导致请求失败。有关更多信息，请参阅 Amazon Simple Storage Service API 参考中的任务 1：创建规范请求。

   如果绝对路径为空，则使用正斜杠 (/)。在示例 IAM 请求中，URI 中的主机后没有任何内容，因此绝对路径为空。

   Example 规范 URI

   ```
   /
   ```

3. 添加规范查询字符串，后跟换行符。如果请求不包括查询字符串，请使用空字符串（实际上是空白行）。示例请求具有以下查询字符串。

   Example 规范查询字符串

   ```
   Action=ListUsers&Version=2010-05-08
   ```

   要构建规范查询字符串，请完成以下步骤：

   a. 按字符代码点以升序顺序对参数名称进行排序。具有重复名称的参数应按值进行排序。例如，以大写字母 F 开头的参数名称排在以小写字母 b 开头的参数名称之前。

   b. 根据以下规则对每个参数名称和值进行 URI 编码：

      - 请勿对 RFC 3986 定义的任何非预留字符进行 URI 编码，这些字符包括：A-Z、a-z、0-9、连字符 (-)、下划线 (_)、句点 (.) 和波形符 (~)。
      - 使用 %XY 对所有其他字符进行百分比编码，其中“X”和“Y”为十六进制字符（0-9 和大写字母 A-F）。例如，空格字符必须编码为 %20（不像某些编码方案那样使用“+”），扩展 UTF-8 字符必须采用格式 `%XY%ZA%BC`。
      - 对参数值中的任何等于 (=) 字符进行双重编码。

   c. 以排序后的列表中第一个参数名称开头，构造规范查询字符串。

   d. 对于每个参数，追加 URI 编码的参数名称，后跟等号字符 (=)，再接 URI 编码的参数值。对没有值的参数使用空字符串。

   e. 在每个参数值后追加与字符 (&)，列表中最后一个值除外。

查询 API 的一种方案是将所有请求参数放入查询字符串中。例如，对 Amazon S3 这样做可以创建预签名 URL。在这种情况下，规范查询字符串不能只包含请求的参数，还必须包含在签名流程中要使用的参数：哈希算法、凭证范围、日期和已签名标头参数。

下面的示例说明包含身份验证信息的查询字符串。该示例中为便于阅读添加了换行符，但在您的代码中，规范查询字符串必须是连续的一行文本。

Example 查询字符串中的身份验证参数

```
Action=ListUsers&
Version=2010-05-08&
X-Amz-Algorithm=AWS4-HMAC-SHA256&
X-Amz-Credential=AKIDEXAMPLE%2F20150830%2Fcn-north-1%2Fiam%2Faws4_request&
X-Amz-Date=20150830T123600Z&
X-Amz-SignedHeaders=content-type%3Bhost%3Bx-amz-date
```

有关身份验证参数的更多信息，请参阅 任务 2：创建 Signature Version 4 的待签字符串 (p. 39)。

Note

您可以使用 Amazon Security Token Service (Amazon STS) 提供的临时安全证书对请求进行签名。其过程与使用长期凭证相同，但在您添加签名信息到查询字符串时，您必须为安全令牌添加额外的查询参数。参数名称为 `X-Amz-Security-Token`，参数值为 URI 编码的会话令牌（在您获取临时安全凭证时从 Amazon STS 收到的字符串）。
对于一些服务，您必须在规范（签名）查询字符串中包括 `X-Amz-Security-Token` 查询参数。对于其他服务，计算签名之后，您需要在末尾添加 `X-Amz-Security-Token` 参数。有关详细信息，请参阅该服务的 API 参考文档。

4. 添加规范标头，后跟换行符。规范标头包括您要包含在签名请求中的所有 HTTP 标头的列表。

对于 HTTP/1.1 请求，您必须至少包含 `host` 标头。标准标头（如 `content-type`）是可选的。对于 HTTP/2 请求，您必须包括 `:authority` 标头，而不是 `host` 标头。不同的服务可能需要其他标头。

Example 规范标头

```
content-type:application/x-www-form-urlencoded; charset=utf-8\n
host:iam.cn-north-1.amazonaws.com.cn\n
x-amz-date:20150830T123600Z\n
```

要创建规范标头列表，请将所有标头名称转换为小写形式并删除前导空格和尾随空格。将标头值中的连续空格转换为单个空格。

以下伪代码描述如何构造规范标头列表：

```
CanonicalHeaders =
CanonicalHeadersEntry0 + CanonicalHeadersEntry1 + ... + CanonicalHeadersEntryN
CanonicalHeadersEntry =
Lowercase(HeaderName) + ':' + Trimall(HeaderValue) + '\n'
```

`Lowercase` 表示将所有字符转换为小写字母的函数。`Trimall` 函数删除值前后的多余空格并将连续空格转换为单个空格。

通过按字符代码对（小写）标头排序，然后对标头名称进行迭代操作，来构建规范标头列表。根据以下规则构造每个标头：

- 追加小写标头名称，后跟冒号。
- 追加该标头的值的逗号分隔列表。请勿对有多个值的标头进行值排序。

- 追加一个新行（“\n”）。

下列示例对更复杂的一组标头及其规范形式进行比较：

Example 原始标头

```
Host:iam.cn-north-1.amazonaws.comcn\n
Content-Type:application/x-www-form-urlencoded; charset=utf-8\n
My-header1:    a   b   c  \n
X-Amz-Date:20150830T123600Z\n
My-Header2:    "a   b   c"  \n
```

Example 规范形式

```
content-type:application/x-www-form-urlencoded; charset=utf-8\n
host:iam.cn-north-1.amazonaws.cn\n
my-header1:a b c\n
my-header2:"a b c"\n
x-amz-date:20150830T123600Z\n
```

> **Note**
>
> 每个标头都后跟换行符，这意味着完整列表以换行符结束。

对于规范形式，进行了下列更改：

- 标头名称已转换为小写字符。
- 标头已按字符代码进行排序。
- 已从 `my-header1` 和 `my-header2` 值中删除前导空格和尾随空格。
- 对于 `my-header1` 和 `my-header2` 值，已将 `a b c` 中的连续空格转换为单个空格。

> **Note**
>
> 您可以使用 Amazon Security Token Service (Amazon STS) 提供的临时安全证书对请求进行签名。该过程与使用长期凭证相同，但在 `Authorization` 标头中包括签名信息时，必须为安全令牌添加额外的 HTTP 标头。标头名称为 `X-Amz-Security-Token`，标头值是会话令牌（在您获取临时安全凭证时从 Amazon STS 收到的字符串）。

5. 添加已签名的标头，后跟换行符。该值是您包含在规范标头中的标头列表。通过添加此标头列表，您可以向Amazon告知请求中的哪些标头是签名过程的一部分，以及在验证请求时Amazon可以忽略哪些标头（例如，由代理添加的任何附加标头）。

   对于 HTTP/1.1 请求，`host` 标头必须作为已签名标头包括在内。对于包含 `:authority` 标头而不是 `host` 标头的 HTTP/2 请求，必须包含 `:authority` 标头作为已签名标头。如果包括日期或 `x-amz-date` 标头，则还必须包括在已签名标头列表中的标头。

   要创建已签名标头列表，请将所有标头名称转换为小写形式，按字符代码对其进行排序，并使用分号来分隔这些标头名称。以下伪代码描述如何构建已签名标头的列表。`Lowercase` 表示将所有字符转换为小写字母的函数。

   ```
   SignedHeaders =
   Lowercase(HeaderName0) + ';' + Lowercase(HeaderName1) + ";" + ... +
    Lowercase(HeaderNameN)
   ```

通过对按小写字符代码排序的标头名称集合进行迭代操作，构建已签名标头的列表。对于除最后一个标头外的每个标头名称，请在标头名称后追加分号（“;”），将它与后面的标头名称分隔开。

Example 已签名标头

```
content-type;host;x-amz-date\n
```

6. 使用 SHA256 等哈希 (摘要) 函数以基于 HTTP 或 HTTPS 请求正文中的有效负载创建哈希值。Signature Version 4 不需要您使用特定字符编码来对有效负载中的文本进行编码。不过，一些 Amazon 服务可能需要特定编码。有关更多信息，请参阅该服务的文档。

   Example 负载结构

   ```
   HashedPayload = Lowercase(HexEncode(Hash(requestPayload)))
   ```

   在创建待签字符串后，请指定用于对有效负载进行哈希处理的签名算法。例如，如果您使用的是 SHA256，则将指定 AWS4-HMAC-SHA256 作为签名算法。经过哈希处理的有效负载必须以小写十六进制字符串形式表示。

   如果有效负载为空，则使用空字符串作为哈希函数的输入。在此 IAM 示例中，有效负载为空。

   Example 经过哈希处理的有效负载（空字符串）

   ```
   e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855
   ```

7. 要构建完整的规范请求，请将来自每个步骤的所有组成部分组合为单个字符串。如上所述，每个组成部分都以换行符结尾。如果您执行前述规范请求伪代码，生成的规范请求将显示在以下示例中。

   Example 规范请求

   ```
   GET
   /
   Action=ListUsers&Version=2010-05-08
   content-type:application/x-www-form-urlencoded; charset=utf-8
   host:iam.cn-north-1.amazonaws.com.cn
   x-amz-date:20150830T123600Z

   content-type;host;x-amz-date
   e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855
   ```

8. 使用您对有效负载进行哈希处理时所使用的相同算法来创建规范请求的摘要（哈希）。

   Note

   在计算摘要之前，签名版本 4 不需要您使用特定字符编码来对规范请求进行编码。不过，一些 Amazon 服务可能需要特定编码。有关更多信息，请参阅该服务的文档。

   经过哈希处理的规范请求必须以小写十六进制字符串形式表示。以下示例显示了使用 SHA-256 对示例规范请求进行哈希处理的结果。

   Example 经过哈希处理的规范请求

   ```
   c0f52cbaae2f5c43042257a73d9eafc6d42e3306a4ebab6d9235f391dff4d990
   ```

   在任务 2：创建 Signature Version 4 的待签字符串 (p. 39)中，将经过哈希处理的规范请求包括到待签字符串中。

任务 2：创建 Signature Version 4 的待签字符串

待签字符串包含有关您的请求和您在任务 1：针对 Signature Version 4 创建规范请求 (p. 34)中创建的规范请求的元信息。您将使用待签字符串和稍后在任务 3：为 Amazon Signature Version 4 计算签名 (p. 40)中为计算请求签名而作为输入创建的派生签名密钥。

要创建待签字符串，请如以下伪代码所示，连接算法、日期和时间、凭证范围和规范请求的摘要：

待签字符串结构

```
StringToSign =
    Algorithm + \n +
    RequestDateTime + \n +
    CredentialScope + \n +
    HashedCanonicalRequest
```

以下示例演示如何使用任务 1：创建规范请求 (p. 34)中的相同请求构造待签字符串。

Example HTTPS 请求

```
GET https://iam.cn-north-1.amazonaws.com.cn/?Action=ListUsers&Version=2010-05-08 HTTP/1.1
Host: iam.cn-north-1.amazonaws.com.cn/
Content-Type: application/x-www-form-urlencoded; charset=utf-8
X-Amz-Date: 20150830T123600Z
```

创建待签字符串

1. 以算法名称开头，后跟换行符。该值是您用于计算规范请求摘要的哈希算法。对于 SHA256，算法是 `AWS4-HMAC-SHA256`。

   ```
   AWS4-HMAC-SHA256\n
   ```

2. 追加请求日期值，后跟换行符。该日期是使用 ISO8601 基本格式以 YYYYMMDD'T'HHMMSS'Z' 格式在 `x-amz-date` 标头中指定的。此值必须与您在前面所有步骤中使用的值匹配。

   ```
   20150830T123600Z\n
   ```

3. 追加凭证范围值，后跟换行符。此值是一个字符串，包含日期、目标区域、所请求的服务和小写字符形式的终止字符串（“`aws4_request`”）。区域和服务名称字符串必须采用 UTF-8 编码。

   ```
   20150830/cn-north-1/iam/aws4_request\n
   ```

   - 日期必须为 `YYYYMMDD` 格式。请注意，日期不包括时间值。
   - 请确保您指定的区域是您将请求发送到的目标区域。

4. 追加您在任务 1：针对 Signature Version 4 创建规范请求 (p. 34)中创建的规范请求的哈希。该值后面不跟换行符。如 RFC 4648 第 8 节所定义，经过哈希处理的规范请求必须为 base-16 编码的小写形式。

   ```
   c0f52cbaae2f5c43042257a73d9eafc6d42e3306a4ebab6d9235f391dff4d990
   ```

以下待签字符串是 2015 年 8 月 30 日对 IAM 的请求。

Example 待签字符串

```
AWS4-HMAC-SHA256
```

```
20150830T123600Z
20150830/cn-north-1/iam/aws4_request
c0f52cbaae2f5c43042257a73d9eafc6d42e3306a4ebab6d9235f391dff4d990
```

任务 3：为 Amazon Signature Version 4 计算签名

在计算签名之前，从 Amazon 秘密访问密钥派生出签名密钥。由于派生签名密钥特定于日期、服务和区域，因此它提供了更高程度的保护。秘密访问密钥不只用于对请求进行签名。然后将签名密钥和您在任务 2：创建 Signature Version 4 的待签字符串 (p. 39)中创建的待签字符串用作加密哈希函数的输入。加密哈希函数生成的十六进制编码结果就是签名。

签名版本 4 不需要您使用特定字符编码来对待签字符串进行编码。不过，一些 Amazon 服务可能需要特定编码。有关更多信息，请参阅该服务的文档。

计算签名

1. 派生您的签名密钥。为此，请使用您的秘密访问密钥创建一系列基于哈希的消息身份验证代码 (HMAC)。此代码显示在以下伪代码中，其中 `HMAC(key, data)` 表示以二进制格式返回输出的 HMAC-SHA256 函数。每个哈希函数的结果将成为下一个函数的输入。

   用于派生签名密钥的伪代码

   ```
   kSecret = your secret access key
   kDate = HMAC("AWS4" + kSecret, Date)
   kRegion = HMAC(kDate, Region)
   kService = HMAC(kRegion, Service)
   kSigning = HMAC(kService, "aws4_request")
   ```

   请注意，哈希过程中所使用的日期的格式为 `YYYYMMDD`（例如，`20150830`），不包括时间。

   确保以正确的顺序为您要使用的编程语言指定 HMAC 参数。在此示例中，密钥是第一个参数，数据 (消息) 是第二个参数，但您使用的函数可能以不同顺序指定密钥和数据。

   使用摘要 (二进制格式) 来派生密钥。大多数语言都有用来计算二进制格式哈希（通常称为摘要）或十六进制编码哈希（称为十六进制摘要）的函数。派生密钥需要使用二进制格式摘要。

   以下示例显示了用于派生签名密钥的输入以及所生成的输出，其中 kSecret = `wJalrXUtnFEMI/K7MDENG+bPxRfiCYEXAMPLEKEY`。

   该示例使用与任务 1 和任务 2 中的请求相同的参数（2015 年 8 月 30 日对 IAM 的请求，位于 `cn-north-1` 区域）。

   示例输入

   ```
   HMAC(HMAC(HMAC(HMAC("AWS4" + kSecret,"20150830"),"cn-north-1"),"iam"),"aws4_request")
   ```

   以下示例显示了此 HMAC 哈希操作序列生成的派生签名密钥。这说明了此二进制签名密钥中每个字节的十六进制表示形式。

   示例签名密钥

   ```
   3fa8337361355535220160ce57f4cb5b8e318209aa7bb03ecdcb9aaeec3d07a2
   ```

   有关如何在不同编程语言中派生签名密钥的更多信息，请参阅说明如何为 Signature Version 4 派生签名密钥的示例 (p. 43)。
2. 计算签名。要计算签名，请使用派生的签名密钥和待签字符串作为加密哈希函数的输入。在计算签名后，将二进制值转换为十六进制表示形式。

以下伪代码说明如何计算签名。

```
signature = HexEncode(HMAC(derived signing key, string to sign))
```

Note

确保以正确的顺序为您要使用的编程语言指定 HMAC 参数。在此示例中，密钥是第一个参数，数据 (消息) 是第二个参数，但您使用的函数可能以不同顺序指定密钥和数据。

以下示例显示了使用与任务 2 中相同的签名密钥和待签字符串会生成的签名：

示例签名

```
d37af66cc90dc26bb2e27d2a97316b729b82589b5e4648f1ae34cb83a3f546cd
```

任务 4：将签名添加到 HTTP 请求

在计算签名后，将它添加到请求。您可以通过以下两种方式之一将签名添加到请求：

- 名为 `Authorization` 的 HTTP 标头
- 查询字符串

您不能同时在 `Authorization` 标头和查询字符串中传递签名信息。

Note

您可以使用 Amazon Security Token Service (Amazon STS) 提供的临时安全证书对请求进行签名。此过程与使用长期凭证相同，但是需要安全令牌的额外 HTTP 标头或查询字符串参数。标头的名称或查询字符串参数为 `X-Amz-Security-Token`，值为会话令牌（在您获取临时安全凭证时从 Amazon STS 收到的字符串）。
将 `X-Amz-Security-Token` 参数添加到查询字符串时，一些服务需要您将此参数包括在规范（签名）请求中。对于其他服务，您在计算签名之后，需要在末尾添加此参数。有关详细信息，请参阅该服务的 API 参考文档。

将签名信息添加到 Authorization 标头

通过将签名信息添加到名为 `Authorization` 的 HTTP 标头，可以包括签名信息。此标头内容是在按前面的步骤所述计算签名之后创建的，因此 `Authorization` 标头未包含在已签名标头的列表中。尽管此标头名为 `Authorization`，但签名信息实际上用于身份验证。

以下伪代码说明 `Authorization` 标头的构造。

```
Authorization: algorithm Credential=access key ID/credential scope,
 SignedHeaders=SignedHeaders, Signature=signature
```

下面的示例说明一个完整的 `Authorization` 标头。

请注意，在实际请求中，Authorization 标头显示为一行连续的文本。为便于阅读，下面的版本已经过格式编排。

```
Authorization: AWS4-HMAC-SHA256
Credential=AKIDEXAMPLE/20150830/cn-north-1/iam/aws4_request,
SignedHeaders=content-type;host;x-amz-date,
```

```
Signature=d37af66cc90dc26bb2e27d2a97316b729b82589b5e4648f1ae34cb83a3f546cd
```

请注意以下几点：

- 算法和 `Credential` 之间没有逗号。但是，`SignedHeaders` 和 `Signature` 使用逗号与之前的值隔开。
- `Credential` 值以访问密钥 ID 开头，后跟正斜杠 (`/`)，再接您在任务 2：创建 Signature Version 4 的待签字符串 (p. 39)中计算得出的凭证值范围。秘密访问密钥用于为签名派生签名密钥，但未包含在通过请求发送的签名信息中。

将签名信息添加到查询字符串

您可以发送请求并在查询字符串中传递所有请求值，包括签名信息。这有时称为预签名 URL，因为它生成单个 URL，其中包含成功调用Amazon所需要的一切信息。通常在 Amazon S3 中使用。有关更多信息，请参阅 Amazon Simple Storage Service API 参考中的使用查询参数（Amazon Signature Version 4）验证请求。

> **Important**
>
> 如果您发出一个请求，其中所有参数都包含在查询字符串中，则生成的 URL 表示已经过身份验证的Amazon操作。因此，对待生成的 URL 要像对待实际凭证一样小心。建议您使用 `X-Amz-Expires` 参数为请求指定较短的过期时间。

如果使用这种方法，所有查询字符串值（签名除外）都将包含在规范查询字符串中，该字符串是您在签名过程第一部分 (p. 34)中构造的规范查询的一部分。

以下伪代码说明包含所有请求参数的查询字符串的构造。

```
querystring = Action=action
querystring += &X-Amz-Algorithm=algorithm
querystring += &X-Amz-Credential= urlencode(access_key_ID + '/' + credential_scope)
querystring += &X-Amz-Date=date
querystring += &X-Amz-Expires=timeout interval
querystring += &X-Amz-SignedHeaders=signed_headers
```

在计算签名（使用其他查询字符串值作为计算的一部分）后，请将该签名作为 `X-Amz-Signature` 参数添加到查询字符串中：

```
querystring += &X-Amz-Signature=signature
```

下面的示例说明当所有请求参数和签名信息都包括在查询字符串参数中时请求看起来是什么样。

请注意，在实际请求中，Authorization 标头显示为一行连续的文本。为便于阅读，下面的版本已经过格式编排。

```
https://iam.cn-north-1.amazonaws.com.cn?Action=ListUsers&Version=2010-05-08
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIDEXAMPLE%2F20150830%2Fcn-north-1%2Fiam%2Faws4_request
&X-Amz-Date=20150830T123600Z
&X-Amz-Expires=60
&X-Amz-SignedHeaders=content-type%3Bhost
&X-Amz-Signature=bbb7890b2172f0cccc6d1d5cded4e690f3e1dac299599547f3d1ceb50567e83d
```

请注意以下几点：

- 为计算签名，查询字符串参数必须按代码点的从低到高的顺序排序，并且其值必须是 URI 编码的值。请参阅任务 1：针对 Signature Version 4 创建规范请求 (p. 34)中有关创建规范查询字符串的步骤。
- 将超时间隔 (`X-Amz-Expires`) 设置为您所请求的操作的最短可行时间。

处理 Signature Version 4 中的日期

您在凭证范围中使用的日期必须与您的请求的日期匹配。您可以用多种方法将日期包括在请求中。您可以使用 `date` 标头或 `x-amz-date` 标头，或者将 `x-amz-date` 作为查询参数包含在内。有关示例请求，请参阅Signature Version 4 完整签名过程的示例（Python）(p. 46)。

时间戳必须采用 UTC 表示，并具有以下 ISO 8601 格式：YYYYMMDD'T'HHMMSS'Z'。例如，`20150830T123600Z` 是有效时间戳。请勿在时间戳中包含毫秒。

Amazon 先检查时间戳的 `x-amz-date` 标头或参数。如果 Amazon 无法找到 `x-amz-date` 的值，则将寻找 `date` 标头。Amazon 检查八位数字字符串形式的凭证范围，表示请求的年 (YYYY)、月 (MM) 和日 (DD)。例如，如果 `x-amz-date` 标头值为 `20111015T080000Z`，并且凭证范围的日期部分为 `20111015`，则 Amazon 允许身份验证过程继续执行。

如果日期不匹配，则 Amazon 拒绝请求，即使时间戳距离凭证范围中的日期仅有数秒之差也是如此。例如，Amazon 将拒绝其 `x-amz-date` 标头值为 `20151014T235959Z` 且凭证范围包括日期 `20151015` 的请求。

说明如何为 Signature Version 4 派生签名密钥的示例

本页以多种编程语言说明示例，介绍如何为 Signature 版本 4 派生签名密钥。此页面上的示例仅说明如何派生签名密钥，它只是 Amazon 请求签名过程的一部分。有关说明整个过程的示例，请参阅Signature Version 4 完整签名过程的示例（Python）(p. 46)。

> **Important**
>
> 如果使用 Amazon SDK 之一（包括 SDK for Java、SDK for .NET、SDK for Python、SDK for Ruby 或 SDK for JavaScript），您不必手动执行派生签名密钥和将身份验证信息添加到请求的步骤。这些软件开发工具包将为您执行这些工作。仅当您直接发出 HTTP 或 HTTPS 请求时，才需要您手动签署请求。

示例

使用 Java 派生签名密钥

```
static byte[] HmacSHA256(String data, byte[] key) throws Exception {
    String algorithm="HmacSHA256";
    Mac mac = Mac.getInstance(algorithm);
    mac.init(new SecretKeySpec(key, algorithm));
    return mac.doFinal(data.getBytes("UTF-8"));
}

static byte[] getSignatureKey(String key, String dateStamp, String regionName, String
 serviceName) throws Exception {
    byte[] kSecret = ("AWS4" + key).getBytes("UTF-8");
    byte[] kDate = HmacSHA256(dateStamp, kSecret);
    byte[] kRegion = HmacSHA256(regionName, kDate);
```

```
    byte[] kService = HmacSHA256(serviceName, kRegion);
    byte[] kSigning = HmacSHA256("aws4_request", kService);
    return kSigning;
}
```

使用 .NET (C#) 派生签名密钥

```
static byte[] HmacSHA256(String data, byte[] key)
{
    String algorithm = "HmacSHA256";
    KeyedHashAlgorithm kha = KeyedHashAlgorithm.Create(algorithm);
    kha.Key = key;

    return kha.ComputeHash(Encoding.UTF8.GetBytes(data));
}

static byte[] getSignatureKey(String key, String dateStamp, String regionName, String
 serviceName)
{
    byte[] kSecret = Encoding.UTF8.GetBytes(("AWS4" + key).ToCharArray());
    byte[] kDate = HmacSHA256(dateStamp, kSecret);
    byte[] kRegion = HmacSHA256(regionName, kDate);
    byte[] kService = HmacSHA256(serviceName, kRegion);
    byte[] kSigning = HmacSHA256("aws4_request", kService);

    return kSigning;
}
```

使用 Python 派生签名密钥

```
def sign(key, msg):
    return hmac.new(key, msg.encode("utf-8"), hashlib.sha256).digest()

def getSignatureKey(key, dateStamp, regionName, serviceName):
    kDate = sign(("AWS4" + key).encode("utf-8"), dateStamp)
    kRegion = sign(kDate, regionName)
    kService = sign(kRegion, serviceName)
    kSigning = sign(kService, "aws4_request")
    return kSigning
```

使用 Ruby 派生签名密钥

```
def getSignatureKey key, dateStamp, regionName, serviceName
    kDate = OpenSSL::HMAC.digest('sha256', "AWS4" + key, dateStamp)
    kRegion = OpenSSL::HMAC.digest('sha256', kDate, regionName)
    kService = OpenSSL::HMAC.digest('sha256', kRegion, serviceName)
    kSigning = OpenSSL::HMAC.digest('sha256', kService, "aws4_request")

    kSigning
end
```

使用 JavaScript (Node.js) 派生签名密钥

以下示例使用 crypto-js 库。有关更多信息，请参阅 https://www.npmjs.com/package/crypto-js 和 https://code.google.com/archive/p/crypto-js/。

```
var crypto = require("crypto-js");
```

```
function getSignatureKey(key, dateStamp, regionName, serviceName) {
    var kDate = crypto.HmacSHA256(dateStamp, "AWS4" + key);
    var kRegion = crypto.HmacSHA256(regionName, kDate);
    var kService = crypto.HmacSHA256(serviceName, kRegion);
    var kSigning = crypto.HmacSHA256("aws4_request", kService);
    return kSigning;
}
```

使用其他语言派生签名密钥

如果您需要用不同编程语言实施此逻辑，我们建议您使用本节中的值来测试密钥派生算法的中间步骤。以下 Ruby 示例在算法的每个步骤后使用 `hexEncode` 函数输出结果。

```
def hexEncode bindata
    result=""
    data=bindata.unpack("C*")
    data.each {|b| result+= "%02x" % b}
    result
end
```

如果使用以下测试输入：

```
key = 'wJalrXUtnFEMI/K7MDENG+bPxRfiCYEXAMPLEKEY'
dateStamp = '20120215'
regionName = 'cn-north-1'
serviceName = 'iam'
```

您的程序将为 `getSignatureKey` 中的值生成以下值。请注意，这些值是二进制数据的十六进制编码表示形式；密钥本身和中间值应该是二进制格式。

```
kSecret  =
 '41575334774a616c725855746e46454d492f4b374d44454e472b6250785266694359455841d504c454b4559'
kDate    = '969fbb94feb542b71ede6f87fe4d5fa29c789342b0f407474670f0c2489e0a0d'
kRegion  = 'f5e672e58cf132b0a7ac38224ed20013b5f068e4e4de6ebc05d87f724508595e'
kService = 'e2569e3d090ed691c9ef28c5fb6afbea3f759699099ad1f884a589aad97bf4ca'
kSigning = '2f93fd817068852310c6054f85a5ffe1a23da3e1587e39ba922f1fac469088da'
```

常见编码错误

要简化您的任务，请避免下列常见编码错误。

> **Tip**
>
> 使用能显示原始 HTTP 请求的工具检查要发送给 Amazon 的 HTTP 请求。这样能帮助您找到代码中并不明显的问题。

- 不要包含多余的换行符，或忘记在必要的位置使用换行符。
- 不要在凭证范围中不正确地设置日期格式，如使用时间戳而不是 YYYYMMDD 格式。
- 确保规范标头和签名标头中的标头相同。
- 不要在计算中间密钥时意外交换密钥和数据 (消息)。上一步的计算结果是密钥，而不是数据。仔细检查文档中的加密基元，确保以正确顺序放置参数。
- 不要忘记在第一步在密钥之前添加字符串“AWS4”。如果使用 `for` 循环或迭代程序来实施密钥派生，不要忘记第一次迭代的特殊情况，以便包含“AWS4”字符串。

有关可能的错误的更多信息，请参阅排除 Amazon 签名版本 4 错误 (p. 53)。

Signature Version 4 完整签名过程的示例（Python）

本节展示用 Python 编写的示例程序，这些程序阐释如何在Amazon中使用 Signature Version 4。我们特意将这些示例程序编写得很简单（仅使用极少 Python 特定功能），便于您更轻松地了解签署Amazon请求的完整过程。

> **Note**
>
> 如果您使用的是 Amazon SDK 之一（包括 SDK for C++、SDK for Go、SDK for Java、Amazon SDK for JavaScript、Amazon SDK for .NET、SDK for PHP、SDK for Python (Boto3) 或 SDK for Ruby），您不必手动执行派生签名密钥和将身份验证信息添加到请求的步骤。这些软件开发工具包将为您执行这些工作。仅当您直接发出 HTTP 或 HTTPS 请求时，才需要您手动签署请求。

要使用这些示例程序，您需要：

- 计算机上安装有 Python 2.x，您可以从 Python 站点获取。这些程序已使用 Python 2.7 和 3.6 测试过。
- Python 请求库，示例脚本使用此库发出 Web 请求。一种方便的 Python 程序包安装方法是使用 `pip`，它可从 Python 程序包索引站点获取程序包。然后，您可以在命令行上运行 `requests` 来安装 `pip install requests`。
- 环境变量中名为 `AWS_ACCESS_KEY_ID` 和 `AWS_SECRET_ACCESS_KEY` 的访问密钥（访问密钥 ID 和私有访问密钥）。或者，您也可以在凭证文件中保存这些值，然后从这些文件中读取。作为最佳实践，我们建议您不要在代码中嵌入凭证。有关更多信息，请参阅亚马逊云科技一般参考中的管理Amazon访问密钥的最佳实践。

以下示例使用 UTF-8 对规范请求和待签字符串进行编码，而签名版本 4 不需要您使用特定字符编码。不过，一些 Amazon 服务可能需要特定编码。有关更多信息，请参阅该服务的文档。

示例

结合使用 GET 和 Authorization 标头 (Python)

以下示例说明了如何在不使用 SDK for Python (Boto3) 的情况下，使用 Amazon EC2 查询 API 发出请求。该请求发出 GET 请求，并使用 `Authorization` 标头将身份验证信息发送到Amazon。

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0

"""
Important

The AWS SDKs sign API requests for you using the access key that you specify when you
configure the SDK. When you use an SDK, you don't need to learn how to sign API requests.
We recommend that you use the AWS SDKs to send API requests, instead of writing your own
 code.

The following example is a reference to help you get started if you have a need to write
your own code to send and sign requests. The example is for reference only and is not
maintained as functional code.
"""

# AWS Version 4 signing example

# EC2 API (DescribeRegions)

# See: http://docs.aws.amazon.com/general/latest/gr/sigv4_signing.html
```

```
# This version makes a GET request and passes the signature
# in the Authorization header.
import sys, os, base64, datetime, hashlib, hmac
import requests # pip install requests

# ************* REQUEST VALUES *************
method = 'GET'
service = 'ec2'
host = 'ec2.amazonaws.com'
region = 'us-east-1'
endpoint = 'https://ec2.amazonaws.com'
request_parameters = 'Action=DescribeRegions&Version=2013-10-15'

# Key derivation functions. See:
# http://docs.aws.amazon.com/general/latest/gr/signature-v4-examples.html#signature-v4-
examples-python
def sign(key, msg):
    return hmac.new(key, msg.encode('utf-8'), hashlib.sha256).digest()

def getSignatureKey(key, dateStamp, regionName, serviceName):
    kDate = sign(('AWS4' + key).encode('utf-8'), dateStamp)
    kRegion = sign(kDate, regionName)
    kService = sign(kRegion, serviceName)
    kSigning = sign(kService, 'aws4_request')
    return kSigning

# Read AWS access key from env. variables or configuration file. Best practice is NOT
# to embed credentials in code.
access_key = os.environ.get('AWS_ACCESS_KEY_ID')
secret_key = os.environ.get('AWS_SECRET_ACCESS_KEY')
if access_key is None or secret_key is None:
    print('No access key is available.')
    sys.exit()

# Create a date for headers and the credential string
t = datetime.datetime.utcnow()
amzdate = t.strftime('%Y%m%dT%H%M%SZ')
datestamp = t.strftime('%Y%m%d') # Date w/o time, used in credential scope


# ************* TASK 1: CREATE A CANONICAL REQUEST *************
# http://docs.aws.amazon.com/general/latest/gr/sigv4-create-canonical-request.html

# Step 1 is to define the verb (GET, POST, etc.)--already done.

# Step 2: Create canonical URI--the part of the URI from domain to query
# string (use '/' if no path)
canonical_uri = '/'

# Step 3: Create the canonical query string. In this example (a GET request),
# request parameters are in the query string. Query string values must
# be URL-encoded (space=%20). The parameters must be sorted by name.
# For this example, the query string is pre-formatted in the request_parameters variable.
canonical_querystring = request_parameters

# Step 4: Create the canonical headers and signed headers. Header names
# must be trimmed and lowercase, and sorted in code point order from
# low to high. Note that there is a trailing \n.
canonical_headers = 'host:' + host + '\n' + 'x-amz-date:' + amzdate + '\n'

# Step 5: Create the list of signed headers. This lists the headers
# in the canonical_headers list, delimited with ";" and in alpha order.
# Note: The request can include any headers; canonical_headers and
# signed_headers lists those that you want to be included in the
# hash of the request. "Host" and "x-amz-date" are always required.
signed_headers = 'host;x-amz-date'
```

```
# Step 6: Create payload hash (hash of the request body content). For GET
# requests, the payload is an empty string ("").
payload_hash = hashlib.sha256(('').encode('utf-8')).hexdigest()

# Step 7: Combine elements to create canonical request
canonical_request = method + '\n' + canonical_uri + '\n' + canonical_querystring + '\n' +
 canonical_headers + '\n' + signed_headers + '\n' + payload_hash


# ************* TASK 2: CREATE THE STRING TO SIGN*************
# Match the algorithm to the hashing algorithm you use, either SHA-1 or
# SHA-256 (recommended)
algorithm = 'AWS4-HMAC-SHA256'
credential_scope = datestamp + '/' + region + '/' + service + '/' + 'aws4_request'
string_to_sign = algorithm + '\n' +  amzdate + '\n' +  credential_scope + '\n' +
 hashlib.sha256(canonical_request.encode('utf-8')).hexdigest()

# ************* TASK 3: CALCULATE THE SIGNATURE *************
# Create the signing key using the function defined above.
signing_key = getSignatureKey(secret_key, datestamp, region, service)

# Sign the string_to_sign using the signing_key
signature = hmac.new(signing_key, (string_to_sign).encode('utf-8'),
 hashlib.sha256).hexdigest()


# ************* TASK 4: ADD SIGNING INFORMATION TO THE REQUEST *************
# The signing information can be either in a query string value or in
# a header named Authorization. This code shows how to use a header.
# Create authorization header and add to request headers
authorization_header = algorithm + ' ' + 'Credential=' + access_key + '/' +
 credential_scope + ', ' +  'SignedHeaders=' + signed_headers + ', ' + 'Signature=' +
 signature

# The request can include any headers, but MUST include "host", "x-amz-date",
# and (for this scenario) "Authorization". "host" and "x-amz-date" must
# be included in the canonical_headers and signed_headers, as noted
# earlier. Order here is not significant.
# Python note: The 'host' header is added automatically by the Python 'requests' library.
headers = {'x-amz-date':amzdate, 'Authorization':authorization_header}


# ************* SEND THE REQUEST *************
request_url = endpoint + '?' + canonical_querystring

print('\nBEGIN REQUEST++++++++++++++++++++++++++++++++++++')
print('Request URL = ' + request_url)
r = requests.get(request_url, headers=headers)

print('\nRESPONSE++++++++++++++++++++++++++++++++++++')
print('Response code: %d\n' % r.status_code)
print(r.text)
```

使用 POST(Python)

以下示例说明了如何在不使用 SDK for Python (Boto3) 的情况下，使用 Amazon DynamoDB 查询 API 发出请求。该请求发出 POST 请求并在请求正文中将值传递给Amazon。身份验证信息是通过 `Authorization` 请求标头传递的。

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0

"""
```

```
Important

The AWS SDKs sign API requests for you using the access key that you specify when you
configure the SDK. When you use an SDK, you don't need to learn how to sign API requests.
We recommend that you use the AWS SDKs to send API requests, instead of writing your own
 code.

The following example is a reference to help you get started if you have a need to write
your own code to send and sign requests. The example is for reference only and is not
maintained as functional code.
"""

# AWS Version 4 signing example

# DynamoDB API (CreateTable)

# See: http://docs.aws.amazon.com/general/latest/gr/sigv4_signing.html
# This version makes a POST request and passes request parameters
# in the body (payload) of the request. Auth information is passed in
# an Authorization header.
import sys, os, base64, datetime, hashlib, hmac
import requests # pip install requests

# ************* REQUEST VALUES *************
method = 'POST'
service = 'dynamodb'
host = 'dynamodb.us-west-2.amazonaws.com'
region = 'us-west-2'
endpoint = 'https://dynamodb.us-west-2.amazonaws.com/'
# POST requests use a content type header. For DynamoDB,
# the content is JSON.
content_type = 'application/x-amz-json-1.0'
# DynamoDB requires an x-amz-target header that has this format:
#     DynamoDB_<API version>.<operationName>
amz_target = 'DynamoDB_20120810.CreateTable'

# Request parameters for CreateTable--passed in a JSON block.
request_parameters =  '{'
request_parameters +=  '"KeySchema": [{"KeyType": "HASH","AttributeName": "Id"}],'
request_parameters +=  '"TableName": "TestTable","AttributeDefinitions": [{"AttributeName":
 "Id","AttributeType": "S"}],'
request_parameters +=  '"ProvisionedThroughput": {"WriteCapacityUnits":
 5,"ReadCapacityUnits": 5}'
request_parameters +=  '}'

# Key derivation functions. See:
# http://docs.aws.amazon.com/general/latest/gr/signature-v4-examples.html#signature-v4-
examples-python
def sign(key, msg):
    return hmac.new(key, msg.encode("utf-8"), hashlib.sha256).digest()

def getSignatureKey(key, date_stamp, regionName, serviceName):
    kDate = sign(('AWS4' + key).encode('utf-8'), date_stamp)
    kRegion = sign(kDate, regionName)
    kService = sign(kRegion, serviceName)
    kSigning = sign(kService, 'aws4_request')
    return kSigning

# Read AWS access key from env. variables or configuration file. Best practice is NOT
# to embed credentials in code.
access_key = os.environ.get('AWS_ACCESS_KEY_ID')
secret_key = os.environ.get('AWS_SECRET_ACCESS_KEY')
if access_key is None or secret_key is None:
    print('No access key is available.')
    sys.exit()
```

```
# Create a date for headers and the credential string
t = datetime.datetime.utcnow()
amz_date = t.strftime('%Y%m%dT%H%M%SZ')
date_stamp = t.strftime('%Y%m%d') # Date w/o time, used in credential scope


# ************* TASK 1: CREATE A CANONICAL REQUEST *************
# http://docs.aws.amazon.com/general/latest/gr/sigv4-create-canonical-request.html

# Step 1 is to define the verb (GET, POST, etc.)--already done.

# Step 2: Create canonical URI--the part of the URI from domain to query
# string (use '/' if no path)
canonical_uri = '/'

## Step 3: Create the canonical query string. In this example, request
# parameters are passed in the body of the request and the query string
# is blank.
canonical_querystring = ''

# Step 4: Create the canonical headers. Header names must be trimmed
# and lowercase, and sorted in code point order from low to high.
# Note that there is a trailing \n.
canonical_headers = 'content-type:' + content_type + '\n' + 'host:' + host + '\n' + 'x-amz-
date:' + amz_date + '\n' + 'x-amz-target:' + amz_target + '\n'

# Step 5: Create the list of signed headers. This lists the headers
# in the canonical_headers list, delimited with ";" and in alpha order.
# Note: The request can include any headers; canonical_headers and
# signed_headers include those that you want to be included in the
# hash of the request. "Host" and "x-amz-date" are always required.
# For DynamoDB, content-type and x-amz-target are also required.
signed_headers = 'content-type;host;x-amz-date;x-amz-target'

# Step 6: Create payload hash. In this example, the payload (body of
# the request) contains the request parameters.
payload_hash = hashlib.sha256(request_parameters.encode('utf-8')).hexdigest()

# Step 7: Combine elements to create canonical request
canonical_request = method + '\n' + canonical_uri + '\n' + canonical_querystring + '\n' +
 canonical_headers + '\n' + signed_headers + '\n' + payload_hash


# ************* TASK 2: CREATE THE STRING TO SIGN*************
# Match the algorithm to the hashing algorithm you use, either SHA-1 or
# SHA-256 (recommended)
algorithm = 'AWS4-HMAC-SHA256'
credential_scope = date_stamp + '/' + region + '/' + service + '/' + 'aws4_request'
string_to_sign = algorithm + '\n' +  amz_date + '\n' +  credential_scope + '\n' +
 hashlib.sha256(canonical_request.encode('utf-8')).hexdigest()

# ************* TASK 3: CALCULATE THE SIGNATURE *************
# Create the signing key using the function defined above.
signing_key = getSignatureKey(secret_key, date_stamp, region, service)

# Sign the string_to_sign using the signing_key
signature = hmac.new(signing_key, (string_to_sign).encode('utf-8'),
 hashlib.sha256).hexdigest()


# ************* TASK 4: ADD SIGNING INFORMATION TO THE REQUEST *************
# Put the signature information in a header named Authorization.
authorization_header = algorithm + ' ' + 'Credential=' + access_key + '/' +
 credential_scope + ', ' +  'SignedHeaders=' + signed_headers + ', ' + 'Signature=' +
 signature
```

```
# For DynamoDB, the request can include any headers, but MUST include "host", "x-amz-date",
# "x-amz-target", "content-type", and "Authorization". Except for the authorization
# header, the headers must be included in the canonical_headers and signed_headers values,
 as
# noted earlier. Order here is not significant.
# # Python note: The 'host' header is added automatically by the Python 'requests' library.
headers = {'Content-Type':content_type,
           'X-Amz-Date':amz_date,
           'X-Amz-Target':amz_target,
           'Authorization':authorization_header}


# ************* SEND THE REQUEST *************
print('\nBEGIN REQUEST++++++++++++++++++++++++++++++++++++')
print('Request URL = ' + endpoint)

r = requests.post(endpoint, data=request_parameters, headers=headers)

print('\nRESPONSE++++++++++++++++++++++++++++++++++++')
print('Response code: %d\n' % r.status_code)
print(r.text)
```

在查询字符串中结合使用 GET 和身份验证信息 (Python)

以下示例说明了如何在不使用 SDK for Python (Boto3) 的情况下，使用 IAM 查询 API 发出请求。该请求发出 GET 请求，并使用查询字符串传递参数和签名信息。

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0

"""
Important

The AWS SDKs sign API requests for you using the access key that you specify when you
configure the SDK. When you use an SDK, you don't need to learn how to sign API requests.
We recommend that you use the AWS SDKs to send API requests, instead of writing your own
 code.

The following example is a reference to help you get started if you have a need to write
your own code to send and sign requests. The example is for reference only and is not
maintained as functional code.
"""

# AWS Version 4 signing example

# IAM API (CreateUser)

# See: http://docs.aws.amazon.com/general/latest/gr/sigv4_signing.html
# This version makes a GET request and passes request parameters
# and authorization information in the query string
import sys, os, datetime, hashlib, hmac, urllib.parse
import requests # pip install requests

# ************* REQUEST VALUES *************
method = 'GET'
service = 'iam'
host = 'iam.amazonaws.com'
region = 'us-east-1'
endpoint = 'https://iam.amazonaws.com'

# Key derivation functions. See:
# http://docs.aws.amazon.com/general/latest/gr/signature-v4-examples.html#signature-v4-
examples-python
def sign(key, msg):
```

```
    return hmac.new(key, msg.encode('utf-8'), hashlib.sha256).digest()

def getSignatureKey(key, dateStamp, regionName, serviceName):
    kDate = sign(('AWS4' + key).encode('utf-8'), dateStamp)
    kRegion = sign(kDate, regionName)
    kService = sign(kRegion, serviceName)
    kSigning = sign(kService, 'aws4_request')
    return kSigning

# Read AWS access key from env. variables or configuration file. Best practice is NOT
# to embed credentials in code.
access_key = os.environ.get('AWS_ACCESS_KEY_ID')
secret_key = os.environ.get('AWS_SECRET_ACCESS_KEY')
if access_key is None or secret_key is None:
    print('No access key is available.')
    sys.exit()

# Create a date for headers and the credential string
t = datetime.datetime.utcnow()
amz_date = t.strftime('%Y%m%dT%H%M%SZ') # Format date as YYYYMMDD'T'HHMMSS'Z'
datestamp = t.strftime('%Y%m%d') # Date w/o time, used in credential scope


# ************* TASK 1: CREATE A CANONICAL REQUEST *************
# http://docs.aws.amazon.com/general/latest/gr/sigv4-create-canonical-request.html

# Because almost all information is being passed in the query string,
# the order of these steps is slightly different than examples that
# use an authorization header.

# Step 1: Define the verb (GET, POST, etc.)--already done.

# Step 2: Create canonical URI--the part of the URI from domain to query
# string (use '/' if no path)
canonical_uri = '/'

# Step 3: Create the canonical headers and signed headers. Header names
# must be trimmed and lowercase, and sorted in code point order from
# low to high. Note trailing \n in canonical_headers.
# signed_headers is the list of headers that are being included
# as part of the signing process. For requests that use query strings,
# only "host" is included in the signed headers.
canonical_headers = 'host:' + host + '\n'
signed_headers = 'host'

# Match the algorithm to the hashing algorithm you use, either SHA-1 or
# SHA-256 (recommended)
algorithm = 'AWS4-HMAC-SHA256'
credential_scope = datestamp + '/' + region + '/' + service + '/' + 'aws4_request'

# Step 4: Create the canonical query string. In this example, request
# parameters are in the query string. Query string values must
# be URL-encoded (space=%20). The parameters must be sorted by name.
canonical_querystring = 'Action=CreateUser&UserName=NewUser&Version=2010-05-08'
canonical_querystring += '&X-Amz-Algorithm=AWS4-HMAC-SHA256'
canonical_querystring += '&X-Amz-Credential=' + urllib.parse.quote_plus(access_key + '/' +
 credential_scope)
canonical_querystring += '&X-Amz-Date=' + amz_date
canonical_querystring += '&X-Amz-Expires=30'
canonical_querystring += '&X-Amz-SignedHeaders=' + signed_headers

# Step 5: Create payload hash. For GET requests, the payload is an
# empty string ("").
payload_hash = hashlib.sha256(('').encode('utf-8')).hexdigest()

# Step 6: Combine elements to create canonical request
```

```
canonical_request = method + '\n' + canonical_uri + '\n' + canonical_querystring + '\n' +
 canonical_headers + '\n' + signed_headers + '\n' + payload_hash


# ************* TASK 2: CREATE THE STRING TO SIGN*************
string_to_sign = algorithm + '\n' +  amz_date + '\n' +  credential_scope + '\n' +
 hashlib.sha256(canonical_request.encode('utf-8')).hexdigest()

# ************* TASK 3: CALCULATE THE SIGNATURE *************
# Create the signing key
signing_key = getSignatureKey(secret_key, datestamp, region, service)

# Sign the string_to_sign using the signing_key
signature = hmac.new(signing_key, (string_to_sign).encode("utf-8"),
 hashlib.sha256).hexdigest()


# ************* TASK 4: ADD SIGNING INFORMATION TO THE REQUEST *************
# The auth information can be either in a query string
# value or in a header named Authorization. This code shows how to put
# everything into a query string.
canonical_querystring += '&X-Amz-Signature=' + signature


# ************* SEND THE REQUEST *************
# The 'host' header is added automatically by the Python 'request' lib. But it
# must exist as a header in the request.
request_url = endpoint + "?" + canonical_querystring

print('\nBEGIN REQUEST++++++++++++++++++++++++++++++++++++')
print('Request URL = ' + request_url)
r = requests.get(request_url)

print('\nRESPONSE++++++++++++++++++++++++++++++++++++')
print('Response code: %d\n' % r.status_code)
print(r.text)
```

排除 Amazon 签名版本 4 错误

在开发实施 Signature Version 4 的代码时，您可能从您测试的Amazon产品收到错误。产生这些错误的原因通常是请求的规范化阶段出现错误、错误地派生或使用了签名密钥，或验证随请求发送的特定于签名的参数失败。

错误

- 排除标准化错误 (p. 53)
- 排除凭证范围错误 (p. 54)
- 排除密钥签名错误 (p. 55)

排除标准化错误

请考虑以下请求：

```
https://iam.cn-north-1.amazonaws.com.cn/?MaxItems=100
&Action=ListGroupsForUser
&UserName=Test
&Version=2010-05-08
&X-Amz-Date=20120223T063000Z
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE/20120223/cn-north-1/iam/aws4_request
&X-Amz-SignedHeaders=host
```

```
&X-Amz-Signature=<calculated value>
```

如果您不正确地计算规范请求或待签字符串，则服务执行的签名验证步骤将失败。以下示例是典型的错误响应，包括规范字符串和服务计算的待签字符串。通过将返回的字符串与规范字符串以及您计算得出的待签字符串进行比较，可以排除您的计算错误。

```
<ErrorResponse xmlns="https://iam.cn-north-1.amazonaws.com.cn/doc/2010-05-08/">
  <Error>
    <Type>Sender</Type>
    <Code>SignatureDoesNotMatch</Code>
    <Message>The request signature we calculated does not match the signature you provided.
 Check your Amazon Secret Access Key and signing method. Consult the service documentation
 for details.

The canonical string for this request should have been 'GET /
 Action=ListGroupsForUser&MaxItems=100&UserName=Test&Version=2010-05-08&X-Amz-
Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential
=AKIAIOSFODNN7EXAMPLE%2F20120223%2Fcn-north-1%2Fiam%2Faws4_request&X-Amz-
Date=20120223T063000Z&X-Amz-SignedHeaders=host
host:iam.cn-north-1.amazonaws.com.cn

host
<hashed-value>'

The String-to-Sign should have been
'AWS4-HMAC-SHA256
20120223T063000Z
20120223/cn-north-1/iam/aws4_request
<hashed-value>'
</Message>
  </Error>
  <RequestId>4ced6e96-5de8-11e1-aa78-a56908bdf8eb</RequestId>
</ErrorResponse>
```

排除凭证范围错误

Amazon 产品可验证凭证范围是否正确；凭证参数必须指定正确的服务、区域和日期。例如，以下凭证引用 Amazon RDS 服务：

```
Credential=AKIAIOSFODNN7EXAMPLE/20120224/cn-north-1/rds/aws4_request
```

如果使用相同的凭证将请求提交到 IAM，您将收到以下错误响应：

```
<ErrorResponse xmlns="https://iam.cn-north-1.amazonaws.com.cn/doc/2010-05-08/">
  <Error>
    <Type>Sender</Type>
    <Code>SignatureDoesNotMatch</Code>
    <Message>Credential should be scoped to correct service: 'iam'. </Message>
  </Error>
  <RequestId>aa0da9de-5f2b-11e1-a2c0-c1dc98b6c575</RequestId>
```

凭证还必须指定正确的区域。例如，IAM 请求的以下凭证错误地指定了美国西部（加利福尼亚北部）区域。

```
Credential=AKIAIOSFODNN7EXAMPLE/20120224/us-west-1/iam/aws4_request
```

```
comma-separated<ErrorResponse xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
  <Error>
    <Type>Sender</Type>
    <Code>SignatureDoesNotMatch</Code>
    <Message>Credential should be scoped to a valid Region, not 'us-west-1'. </Message>
```

```
  </Error>
  <RequestId>8e229682-5f27-11e1-88f2-4b1b00f424ae</RequestId>
</ErrorResponse>
```

如果您向凭证范围指定区域之外的其他区域提交请求，则您将从多个区域中的 Amazon 产品收到相同类型的无效区域响应。

凭证还必须为请求中的服务和操作指定正确的区域。

用作证书组成部分的日期必须与 x-amz-date 标头中的日期值匹配。例如，以下 x-amz-date 标头值与它之后的 Credential 参数中使用的日期值不匹配。

```
x-amz-date:"20120224T213559Z"
Credential=AKIAIOSFODNN7EXAMPLE/20120225/cn-north-1/iam/aws4_request
```

如果您使用此 x-amz-date 标头和证书组合，则会收到以下错误响应：

```
<ErrorResponse xmlns="https://iam.cn-north-1.amazonaws.com.cn/doc/2010-05-08/">
  <Error>
    <Type>Sender</Type>
    <Code>SignatureDoesNotMatch</Code>
    <Message>Date in Credential scope does not match YYYYMMDD from ISO-8601 version of date
 from HTTP: '20120225' != '20120224', from '20120 224T213559Z'.</Message>
  </Error>
  <RequestId>9d6ddd2b-5f2f-11e1-b901-a702cd369eb8</RequestId>
</ErrorResponse>
```

过期的签名也可能生成错误响应。例如，以下错误响应是由于签名过期产生的。

```
<ErrorResponse xmlns="https://iam.cn-north-1.amazonaws.com.cn/doc/2010-05-08/">
  <Error>
    <Type>Sender</Type>
    <Code>SignatureDoesNotMatch</Code>
    <Message>Signature expired: 20120306T074514Z is now earlier than 20120306T074556Z
 (20120306T080056Z - 15 min.)</Message>
  </Error>
  <RequestId>fcc88440-5dec-11e1-b901-a702cd369eb8</RequestId>
</ErrorResponse>
```

排除密钥签名错误

由于不正确地派生签名密钥或使用密码术而导致的错误更难排除。错误响应还会告诉您签名不匹配。如果您已验证规范字符串和待签字符串正确，则签名不匹配的原因很可能是以下两个问题之一：

- 私有访问密钥与您在 Credential 参数中指定的访问密钥 ID 不匹配。
- 您的密钥派生代码存在问题。

要检查私有密钥是否与访问密钥 ID 匹配，可将私有密钥和访问密钥 ID 用于已知可工作的实施。一种方法是使用 Amazon SDK 之一来编写程序，在程序中使用相关访问密钥 ID 和秘密访问密钥向Amazon发出简单请求。

要检查您的密钥派生代码是否正确，您可将它与我们的示例派生代码进行对比。有关更多信息，请参阅说明如何为 Signature Version 4 派生签名密钥的示例 (p. 43)。

Signature Version 4 的服务特定参考

要了解有关在特定的 Amazon 服务上下文中发出 HTTP 请求并对请求进行签名的信息，请参阅以下服务的文档：

- Amazon API Gateway
- Amazon CloudSearch
- Amazon CloudWatch
- Amazon Data Pipeline
- Amazon Elastic Compute Cloud (Amazon EC2)
- Amazon Elastic Transcoder
- Amazon S3 Glacier
- Amazon Mobile Analytics
- Amazon Relational Database Service (Amazon RDS)
- Amazon Simple Email Service (Amazon SES)
- Amazon Simple Queue Service (Amazon SQS)
- Amazon Simple Storage Service (Amazon S3)
- Amazon Simple Workflow Service (Amazon SWF)
- Amazon WAF

Signature Version 2 签名流程

Important

Amazon SDK、Amazon Command Line Interface (Amazon CLI) 和其他 Amazon 工具会使用您在配置工具时指定的访问密钥为您签署 API 请求。当您使用这些工具时，您不必了解如何签署这些 API 请求。以下文档说明了如何签署 API 请求，但仅适用于您编写自己的代码来发送和签署 Amazon API 请求的情况。建议使用 Amazon SDK 或其他 Amazon 工具来发送 API 请求，而不是编写自己的代码。
如果您必须编写自己的代码来签署 Amazon API 请求，请使用 Signature Version 4（SigV4）(p. 30)。

受支持的区域和服务

您可以使用 Signature Version 2 为某些 Amazon 区域中的某些 Amazon 服务签署 API 请求。否则，必须使用 Signature Version 4 来签署 API 请求。

支持 Signature Version 2 的区域

- 美国东部（弗吉尼亚州北部）区域
- 美国西部（加利福尼亚北部）区域
- 美国西部（俄勒冈州）区域
- 欧洲（爱尔兰）区域
- 亚太区域（东京）
- 亚太区域（新加坡）
- 亚太区域（悉尼）
- 南美洲（圣保罗）区域

支持 Signature Version 2 的服务

- Amazon EC2 Auto Scaling
- Amazon CloudFormation
- Amazon CloudWatch

- Amazon Elastic Beanstalk
- Amazon Elastic Compute Cloud (Amazon EC2)
- Elastic Load Balancing
- Amazon EMR
- Amazon ElastiCache
- Amazon Identity and Access Management (IAM)
- Amazon Import/Export
- Amazon Relational Database Service (Amazon RDS)
- Amazon Simple Notification Service (Amazon SNS)
- Amazon Simple Queue Service (Amazon SQS)
- Amazon SimpleDB

弃用 Signature Version 2 的服务

- Amazon Simple Storage Service (Amazon S3) – Amazon S3 更新 – 弃用 SigV2
- Amazon Simple Email Service (Amazon SES)

Signature Version 2 的查询请求组件

Amazon 要求采用签名版本 2 格式的每个 HTTP 或 HTTPS 查询请求都包含以下内容：

端点

也称为 HTTP 请求的主机部分。这是您发送查询请求的计算机的 DNS 名称。它对于每个 Amazon 区域是不同的。

操作

您希望 Web 服务执行的操作。这个值用于确定在请求中使用的参数。

AWSAccessKeyId

注册Amazon账户时由Amazon分发的值。

签名方法

用于计算签名的基于哈希的协议。它可以是适用于签名版本 2 的 HMAC-SHA1 或 HMAC-SHA256。

签名版本

Amazon签名协议的版本。

时间戳

您发出请求的时间。在查询请求中包含此值有助于防止第三方截取您的请求。

必需和可选参数

每个操作都有一组用于定义 API 调用的必需参数和可选参数。

签名

计算得出的值，用于确保签名有效和未被篡改。

下面是一个采用 HTTPS GET 请求格式的示例 Amazon EMR 查询请求。

- 终端节点 `elasticmapreduce.amazonaws.com` 是默认终端节点，它映射到区域 us-east-1。

- 操作是 DescribeJobFlows，它请求有关一个或多个任务流程的信息。

  Note

  实际的查询请求中没有空格或换行符。请求是连续的文本行。下面的版本已经进行了格式设置，便于阅读。

```
https://elasticmapreduce.amazonaws.com?
&AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE
&Action=DescribeJobFlows
&SignatureMethod=HmacSHA256
&SignatureVersion=2
&Timestamp=2011-10-03T15%3A19%3A30
&Version=2009-03-31
&Signature=calculated value
```

如何生成查询请求的签名

Web 服务请求通过 Internet 发送，易被篡改。为了确保请求未被更改，Amazon会计算签名，以确定任何参数或参数值在传输途中是否发生了更改。Amazon要求把签名作为每个请求的一部分。

请确保对该请求进行 URI 编码。例如，您请求中的空白应编码为 %20。虽然 HTTP 协议规范通常允许未编码的空格，但使用未编码的字符会使查询请求中的签名无效。请勿将空格编码为加号 (+)，因为这会导致错误。

以下主题将介绍使用 Amazon Signature Version 2 计算签名所需的步骤。

任务 1：设置查询请求的格式

在对查询请求签名之前，请将请求设置为标准化 (规范) 格式。这是因为采用不同的方式设置查询请求格式会得到不同的 HMAC 签名。在签名前请将请求设置为规范格式。这可以确保您的应用程序和Amazon将为请求计算出相同的签名。

要创建待签字符串，您需要连接查询请求组件。下面的示例为以下 Amazon EMR API 调用生成待签字符串。

```
https://elasticmapreduce.amazonaws.com?
Action=DescribeJobFlows
&Version=2009-03-31
&AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE
&SignatureVersion=2
&SignatureMethod=HmacSHA256
&Timestamp=2011-10-03T15:19:30
```

Note

前面请求中的最后四个参数（从 AWSAccessKeyID 到 Timestamp）称为身份验证参数。每个 Signature Version 2 请求中都需要这些参数。Amazon 使用它们来确定谁发送请求以及是否授予请求的访问权限。

创建待签字符串

1. 先是请求方法（GET 或 POST），然后是换行符。为便于阅读，换行符表示为 \n。

```
GET\n
```

2. 添加小写的 HTTP 主机标头（终端节点），然后添加换行符。若为协议的标准端口（HTTP 端口 80 和 HTTPS 端口 443），可以省略端口信息；若为非标准端口，则需包含端口信息。

```
elasticmapreduce.amazonaws.com\n
```

3. 添加 URI 的每个路径分段的 URL 编码版本（指的是 HTTP 主机标头到查询字符串参数开头的问号字符 (?) 之间的全部字符），后面加换行符。请勿编码用于划定每个路径分段的正斜杠 (/)。

   在本例中，如果绝对路径为空，请使用正斜杠 (/)。

```
/\n
```

4. a. 添加查询字符串组件，使用 UTF-8 字符形式，它们应进行了 URL 编码（十六进制字符必须大写）。您没有对请求中的初始问号字符 (?) 进行编码。有关更多信息，请参见 RFC 3986。
   b. 按字节顺序对查询字符串组件排序。字节顺序区分大小写。Amazon根据原始字节对这些组件进行排序。

      例如，下面是查询字符串组件的原始顺序。

```
Action=DescribeJobFlows
Version=2009-03-31
AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE
SignatureVersion=2
SignatureMethod=HmacSHA256
Timestamp=2011-10-03T15%3A19%3A30
```

      这些查询字符串组件会重新组织为以下顺序：

```
AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE
Action=DescribeJobFlows
SignatureMethod=HmacSHA256
SignatureVersion=2
Timestamp=2011-10-03T15%3A19%3A30
Version=2009-03-31
```

   c. 使用等号字符 (=) 将参数名称与参数值分离，即使参数值为空也如此。使用与字符 (&) 分隔参数和值对。将参数及其值连接组成一个长字符串，中间没有空格。允许参数值内有空格，但空格必须经 URL 编码成 %20。在连接后的字符串中，句点字符 (.) 未进行转义。RFC 3986 将句点字符视为非保留字符，因此未对其进行 URL 编码。

      Note

      RFC 3986 没有指定对 ASCII 控制字符、扩展的 UTF-8 字符以及 RFC 1738 预留的其他字符作何处理。由于任何值都可能成为字符串值，这些其他字符应该进行百分数编码为 %XY，其中 X 和 Y 为大写的十六进制字符。扩展的 UTF-8 字符采用 %XY%ZA... 的形式（可以处理多字节）。

   下面的示例介绍查询字符串组件，参数采用与字符 (&) 连接，并按字节顺序排序。

```
AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&Action=DescribeJobFlows&SignatureMethod=HmacSHA256&SignatureVer
```

5. 要构建最终规范请求，请将每个步骤的所有组成部分组合起来。如下所示，每个组成部分都以换行符结尾。

```
GET\n
elasticmapreduce.amazonaws.com\n
/\n
AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&Action=DescribeJobFlows&SignatureMethod=HmacSHA256&SignatureVer
```

任务 2：计算签名

按任务 1：设置查询请求的格式 (p. 58)所述创建规范字符串后，使用 HMAC-SHA1 或 HMAC-SHA256 协议创建基于哈希的消息身份验证代码 (HMAC)，然后计算签名。建议使用 HMAC-SHA256 协议。

在本例中，签名是使用以下规范字符串和私有密钥作为加密哈希函数的输入而计算的：

- 规范查询字符串：

```
GET\n
elasticmapreduce.amazonaws.com\n
/\n
AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&Action=DescribeJobFlows&SignatureMethod=HmacSHA256&SignatureVersi
```

- 示例私有密钥：

```
wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
```

所得签名必须采用 Base-64 编码。

```
i91nKc4PWAt0JJIdXwz9HxZCJDdiy6cf%2FMj6vPxyYIs%3D
```

将得到的值作为 `Signature` 参数添加到查询请求中。在将此参数添加到请求时，您必须像对任何其他参数一样对此参数进行 URI 编码。您可以在 HTTP 或 HTTPS 调用中使用已签名的请求。

```
https://elasticmapreduce.amazonaws.com?
AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&Action=DescribeJobFlows&SignatureMethod=HmacSHA256&SignatureVersion
%2FMj6vPxyYIs%3D
```

> **Note**
>
> 您可以使用 Amazon Security Token Service (Amazon STS) 提供的临时安全证书对请求进行签名。此过程与使用长期凭证相同，但是请求需要安全令牌的其他参数。

以下请求使用临时访问密钥 ID 和 `SecurityToken` 参数。

Example 具有临时安全凭证的示例请求

```
https://sdb.amazonaws.com/
?Action=GetAttributes
&AWSAccessKeyId=access-key-from-Amazon Security Token Service
&DomainName=MyDomain
&ItemName=MyItem
&SignatureVersion=2
&SignatureMethod=HmacSHA256
&Timestamp=2010-01-25T15%3A03%3A07-07%3A00
&Version=2009-04-15
&Signature=signature-calculated-using-the-temporary-access-key
&SecurityToken=session-token
```

有关更多信息，请参阅以下 资源：

- Amazon EMR 开发人员指南中包含有关 Amazon EMR API 调用的信息。
- 各服务的 API 文档中包含有关操作要求和特定参数的信息。
- Amazon 软件开发工具包提供用以生成查询请求签名的函数。要查看使用Amazon SDK for Java的示例，请参阅使用 Java 开发工具包签署查询请求 (p. 61)。

排查请求签名的问题

本部分描述了您在最初开发代码生成签名以签署查询请求时可能看到的一些错误代码。

Web 服务中的 SignatureDoesNotMatch 签名错误

如果 Web 服务尝试通过重新计算签名值来验证请求签名，但生成的值与附加到请求的签名不匹配，将返回以下错误响应。这可能是因为在发送请求和请求到达 Web 服务终端节点的过程中，请求被修改（这就是签名要检测的情况），或者是因为签名计算错误。以下错误消息的常见原因是不恰当地创建了待签字符串，例如，忘记对 Amazon S3 存储桶名称中的一些字符进行 URL 编码，如冒号（:）和正斜杠（/）。

```
<ErrorResponse xmlns="http://elasticmapreduce.amazonaws.com/doc/2009-03-31">
  <Error>
    <Type>Sender</Type>
    <Code>SignatureDoesNotMatch</Code>
    <Message>The request signature we calculated does not match the signature you
 provided.
    Check your Amazon Secret Access Key and signing method.
    Consult the service documentation for details.</Message>
  </Error>
  <RequestId>7589637b-e4b0-11e0-95d9-639f87241c66</RequestId>
</ErrorResponse>
```

Web 服务中的 IncompleteSignature 签名错误

以下错误表明该签名缺少信息或格式不正确。

```
<ErrorResponse xmlns="http://elasticmapreduce.amazonaws.com/doc/2009-03-31">
  <Error>
    <Type>Sender</Type>
    <Code>IncompleteSignature</Code>
    <Message>Request must contain a signature that conforms to Amazon standards</Message>
  </Error>
  <RequestId>7146d0dd-e48e-11e0-a276-bd10ea0cbb74</RequestId>
</ErrorResponse>
```

使用 Java 开发工具包签署查询请求

以下示例使用 Amazon SDK for Java 的 amazon.webservices.common 软件包来生成 Amazon Signature Version 2 查询请求签名。要做到这一点，这个包创建符合 RFC 2104 要求的 HMAC 签名。有关 HMAC 的更多信息，请参阅 HMAC：用于消息身份验证的哈希密钥。

> **Note**
>
> Java 用作示例实施。您可以使用所选的编程语言实施 HMAC 算法以签署查询请求。

```
import java.security.SignatureException;
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
```

```
import com.amazonaws.util.*;

/**
* This class defines common routines for generating
* authentication signatures for Amazon Platform requests.
*/
public class Signature {
    private static final String HMAC_SHA256_ALGORITHM = "HmacSHA256";

    /**
     * Computes RFC 2104-compliant HMAC signature.
     * * @param data
     * The signed data.
     * @param key
     * The signing key.
     * @return
     * The Base64-encoded RFC 2104-compliant HMAC signature.
     * @throws
     * java.security.SignatureException when signature generation fails
     */
    public static String calculateRFC2104HMAC(String data, String key)
    throws java.security.SignatureException
    {
        String result;
        try {

            // Get an hmac_sha256 key from the raw key bytes.
            SecretKeySpec signingKey = new SecretKeySpec(key.getBytes("UTF-8"),
 HMAC_SHA256_ALGORITHM);

            // Get an hmac_sha256 Mac instance and initialize with the signing key.
            Mac mac = Mac.getInstance(HMAC_SHA256_ALGORITHM);
            mac.init(signingKey);

            // Compute the hmac on input data bytes.
            byte[] rawHmac = mac.doFinal(data.getBytes("UTF-8"));

            // Base64-encode the hmac by using the utility in the SDK
            result = BinaryUtils.toBase64(rawHmac);


        } catch (Exception e) {
            throw new SignatureException("Failed to generate HMAC : " + e.getMessage());
        }
        return result;
    }
}
```

Amazon 软件开发工具包支持 Amazon S3 客户端加密

下表列出了语言特定的 Amazon SDK 支持的加密算法和功能。有关如何使用特定 SDK 的功能的信息，请参阅该 SDK 的开发人员指南。

如果您是初次使用加密，请参阅《Amazon 加密服务和工具指南》了解术语和概念。

Note

Amazon Encryption SDK 是独立于 Amazon SDK 的客户端加密库。您可以使用此加密库更轻松地实施加密最佳做法。与特定于语言的 Amazon SDK 中的 Amazon S3 加密客户端不同，Amazon

Encryption SDK 将返回可移植加密文字，它不与 Amazon S3 绑定，不需要 Amazon Web Services 账户，并可用于加密或解密任何未格式化数据。
Amazon Encryption SDK 和 Amazon S3 加密客户端不兼容，因为它们生成具有不同数据格式的密文。有关 Amazon Encryption SDK 的更多信息，请参阅 Amazon Encryption SDK 开发人员指南。

用于 Amazon S3 客户端加密的 Amazon SDK 功能

要在上传到 Amazon S3 之前使用 Amazon S3 客户端加密库对数据进行加密，您必须向 Amazon S3 加密客户端提供根密钥。您可以提供客户端根密钥，或者使用来自 Amazon Key Management Service (Amazon KMS) 的 Amazon KMS 密钥。Amazon KMS 密钥可使安全地创建和管理加密密钥变得更加轻松。有关这些功能的更多信息，请选择功能列中提供的链接。

有关如何使用特定软件开发工具包的功能的详细信息，请参阅该软件开发工具包的开发人员指南。

在下表中，每一列指示特定语言的 Amazon Command Line Interface 或 SDK 是否支持客户端加密中使用的功能。

功能	Java	.NET	Ruby v2	Amazon CLI	Boto3	PHP v3	JavaScrip	Go	C++
Amazon S3 客户端加密	是	是	是	否	否	是	否	是	是
Amazon KMS 密钥	是	是	是	否	否	是	否	是	是

有关支持客户端加密的 v2 Amazon S3 加密客户端的信息，请参阅我们的博客文章 Updates to the Amazon S3 Encryption Client。

有关旧版 v1 Amazon S3 加密客户端的更多详细信息，请参阅以下博客文章。

- 使用 Amazon SDK for Java 的适用于 Amazon S3 的客户端数据加密
- 使用 Amazon SDK for .NET 和 Amazon S3 的客户端数据加密
- 在 Amazon SDK for Ruby 中使用适用于 Amazon S3 的客户端加密
- 使用 Amazon SDK for Go 加密客户端
- 面向 C++ 开发人员的 Amazon S3 加密客户端现已推出

Amazon S3 加密客户端加密算法

下表列出了使用 Amazon S3 加密客户端时，每个特定语言的 Amazon SDK 对加密密钥和数据支持的算法列表。

算法	Java	.NET	Ruby v2	Amazon CLI	Boto3	PHP v3	JavaScrip	Go	C++
密钥包装：RSA-OAEP-SHA1	是	是	是	否	否	否	否	否	否

算法	Java	.NET	Ruby v2	Amazon CLI	Boto3	PHP v3	JavaScrip	Go	C++
密钥包装：AES/GCM	是	是	是	否	否	否	否	否	是
密钥包装：KMS + 上下文	是	是	是	否	否	是	否	是	是
密钥包装：AES/ECB	弃用	弃用	弃用	否	否	否	否	否	否
密钥包装：AESWrap	弃用	弃用	弃用	否	否	否	否	否	弃用
密钥包装：RSA	弃用	否	弃用	否	否	否	否	否	否
密钥包装：KMS	弃用	弃用	弃用	否	否	弃用	否	弃用	弃用
内容加密：AES/GCM	是	是	是	否	否	是	否	是	是
内容加密：AES/CBC	弃用	否	弃用	否	否	否	否	弃用	弃用

有关经身份验证和仅加密模式的更多信息，请参阅博客文章 Amazon S3 Client-Side Authenticated Encryption。

文档惯例

以下是 Amazon 技术出版物的常见印刷惯例。

内联代码 (例如命令、操作、参数、常数、XML 元素和正则表达式)

格式：文本采用等宽字体

示例：`java -version`

示例数据块 (例如，示例代码和脚本)

格式：阴影块中的文本，采用等宽字体

例如：

```
# ls -l /var/www/html/index.html
-rw-rw-r--  1 root root 1872 Jun 21 09:33 /var/www/html/index.html
# date
Wed Jun 21 09:33:42 EDT 2006
```

互斥选项

格式：用竖线分隔的文本

示例：`(start | stride | edge)`

可选参数

格式：文本用方括号括起

示例：`[-n, -quiet]`

定义

格式：斜体文本

示例：Amazon Machine Image（AMI）

技术出版物

格式：斜体文本

示例：《Amazon Simple Storage Service 用户指南》

用户界面中的元素

格式：粗体文本

示例：选择文件、属性。

用户输入 (用户键入的文本)

格式：文本采用等宽字体

示例：对于名称，键入 **`my-new-resource`**。

必需值的占位符文本

格式：*##* 文本

例如：

```
aws ec2 register-image --image-location my-s3-bucket/image.manifest.xml
```

Amazon词汇表

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) | O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

数字和符号

100-continue	一种方法，使客户能够在实际发送请求之前，查看服务器是否可以接受请求。对于大型的 PUT 请求而言，这种方法可以同时节省时间和带宽费用。

A

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) | O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

AAD	See 其他已经过身份验证的数据.
访问分析器	Amazon Identity and Access Management (IAM) (p. 78) 的一项功能，您可以使用该功能来识别组织和账户中与外部实体共享的资源。示例资源包括 Amazon S3 存储桶或 IAM 角色。 See Also https://aws.amazon.com/about-aws/whats-new/2019/12/introducing-aws-identity-and-access-management-access-analyzer/.
访问控制列表 (ACL)	定义哪些人可以访问特定桶 (p. 83)或对象的文档。Amazon S3 (p. 74) 中的每个桶 (p. 83)和对象都有一个 ACL。此文档定义每个类型的用户可以执行哪些操作，如写入和读取权限。
访问标识符	See 凭证.
访问密钥	访问密钥 ID (p. 67)（例如，`AKIAIOSFODNN7EXAMPLE`）和 私有访问密钥 (p. 109)（例如，`wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`）的组合。使用访问密钥来签署您向 Amazon 发出的 API 请求。
访问密钥 ID	与 私有访问密钥 (p. 109) 关联的唯一标识符；访问密钥 ID 和秘密访问密钥一起使用，对采用编程方式的 Amazon 请求进行加密签名。
访问密钥交替	一种通过更改 Amazon 访问密钥 ID 来提高安全性的方法。您可以使用此方法自行决定停用旧密钥。

访问策略语言	一种用于编写指定哪些人在哪些条件下可以访问特定 Amazon 资源 (p. 106)的文档（具体是策略 (p. 103)）的语言。
账户	与 Amazon 的正式关系，它与以下所有内容关联： • 所有者电子邮件地址和密码 • 对在其伞形结构下创建的资源的控制 • 对与这些资源相关的 Amazon 活动的付款 Amazon Web Services 账户 有权使用所有 Amazon Web Services 账户 资源执行一切操作。这与user (p. 114)正好相反，用户是账户中包含的实体。
账户活动	用于显示您月初至今 Amazon 使用情况和费用的网页。账户活动页面网址为 https://aws.amazon.com/account-activity/。
ACL	See 访问控制列表 (ACL).
ACM	See Amazon Certificate Manager (ACM).
ACM PCA	See Amazon Private Certificate Authority (ACM PCA).
ACM Private CA	See Amazon Private Certificate Authority (ACM PCA).
action	一个 API 函数。也称为操作或调用。委托人 (p. 104) 有权执行的活动。操作是“如果 D 适用的情况下，那么 A 可以对 C 执行 B”语句中的 B。例如，Jane 使用 `Action=ReceiveMessage` 向 Amazon SQS (p. 73) 发送请求。 Amazon CloudWatch (p. 69)：警报状态的变化（例如从 `OK` 变为 `ALARM`）导致的响应。状态变化可能是由于某个指标达到警报阈值引起的，或者是由 `SetAlarmState` 请求引起的。每个警报都可能对每个状态指定一个或多个动作。每当警报的状态变为具有已分配操作时，就会执行操作一次。示例操作包括 Amazon Simple Notification Service (p. 73) 通知，运行 Amazon EC2 Auto Scaling (p. 70) 策略 (p. 103)，以及 Amazon EC2 (p. 70) 实例 (p. 95) 停止/终止操作。
活动可信密钥组	一个列表，其中显示了在 Amazon CloudFront 中处于活动状态可进行分配的每个 可信密钥组 (p. 114) 以及每个密钥组中的公有密钥 ID。CloudFront 可以使用这些密钥组中的公有密钥验证 CloudFront 签名 URL 和签名 Cookie 的签名。
有效的可信签署人	请参阅活动可信密钥组 (p. 68)。
其他已经过身份验证的数据	已检查完整性但未加密的信息，例如标头或其他上下文元数据。
管理暂停	Amazon EC2 Auto Scaling (p. 70) 可能会暂停多次启动实例失败的 自动扩缩组 (p. 75) 的流程。最常遇到管理暂停的 Auto Scaling 组通常符合以下条件：没有正在运行的实例，持续尝试启动实例的时间超过 24 小时，但是在这段时间内未成功启动。
告警	一个项目，其在指定时间段内监控单个指标并启动 Amazon SNS (p. 73) topic (p. 114) 或者 Amazon EC2 Auto Scaling (p. 70) 策略 (p. 103)。如果在一个预先确定的若干个时间段内指标值超过了阈值，则启动这些操作。
允许	评估 IAM (p. 78) 访问策略 (p. 103)时的两种可能结果之一（另一个是拒绝 (p. 88)）。用户向 Amazon 提出请求时，Amazon 将基于应用于用户的所有权限评估请求，然后返回允许或拒绝。
Amazon API Gateway	一项完全托管式服务，开发人员可以使用该服务创建、发布、维护、监控和保护任何规模的 API。 See Also https://aws.amazon.com/api-gateway.

Amazon AppStream 2.0	一个完全托管的安全服务，可将桌面应用程序流式传输给用户，而无需重写这些应用程序。 See Also https://aws.amazon.com/appstream/.
Amazon Athena	一项交互式查询服务，您可以使用该服务通过 ANSI SQL 在 Amazon S3 中分析数据。Athena 是无服务器服务，因此无需管理基础设施。Athena 会自动扩展且易于使用，因此，您可以在几秒钟内开始分析数据集。 See Also https://aws.amazon.com/athena/.
Amazon Aurora	一个完全托管的、与 MySQL 兼容的关系数据库引擎，它结合了商用数据库的速度和可用性，同时还具有开源数据库的简单性和成本效益。 See Also https://aws.amazon.com/rds/aurora/.
Amazon Chime	一项安全的实时统一通信服务，使会议更加高效和易于举行，从而带来会议的转变。 See Also https://aws.amazon.com/chime/.
Amazon Cloud Directory (Cloud Directory)	为应用程序的多层次数据提供高度可扩展的目录存储的服务。 See Also https://aws.amazon.com/cloud-directory/.
Amazon CloudFront	一项 Amazon 内容分发服务，可帮助您提高网站和应用程序的性能、可靠性和可用性。 See Also https://aws.amazon.com/cloudfront.
Amazon CloudSearch	一项完全托管在 Amazon Web Services 云 中的服务，您可以使用该服务为网站或应用程序设置、管理和扩展搜索解决方案。
Amazon CloudWatch	一项 Web 服务，可用于监控和管理各种指标，并根据这些指标的数据配置警报操作。 See Also https://aws.amazon.com/cloudwatch.
Amazon CloudWatch Events	一项 Web 服务，您可以使用该服务将描述 Amazon 资源 (p. 106)中变化的即时系统事件流传输到 Amazon Lambda (p. 79) 函数、Amazon Kinesis Data Streams (p. 71) 中的流、Amazon Simple Notification Service (p. 73) 主题或内置目标。 See Also https://aws.amazon.com/cloudwatch.
Amazon CloudWatch Logs	一项 Web 服务，用于从现有的系统、应用程序和自定义日志文件监控您的系统和应用程序并进行问题排查。您可以将现有日志文件发送到 CloudWatch Logs，并以近乎实时的方式监控这些日志。 See Also https://aws.amazon.com/cloudwatch.
Amazon Cognito	一项 Web 服务，您可以使用该服务将移动用户数据保存在 Amazon Web Services 云 中，无需编写任何后端代码或管理任何基础设施。您可以保存的移动用户数据示例包括应用程序首选项和游戏状态。Amazon Cognito 支持跨设备执行移动身份管理和数据同步。 See Also https://aws.amazon.com/cognito/.
Amazon Comprehend	一项自然语言处理（NLP）服务，通过机器学习发现文本中的见解和关系。 See Also https://aws.amazon.com/comprehend/.
Amazon Comprehend Medical	一项符合 HIPAA 要求的自然语言处理（NLP）服务，通过机器学习从医疗文本中提取健康数据。 See Also https://aws.amazon.com/comprehend/medical/.
Amazon Connect	一项服务解决方案，提供自助配置，支持任何规模的动态、个人和自然的客户互动。 See Also https://aws.amazon.com/connect/.
Amazon Corretto	开放 Java 开发工具包 (OpenJDK) 的免费、多平台、生产就绪型分发版。 See Also https://aws.amazon.com/corretto/.

Amazon Detective	一项服务，该服务从您的 Amazon 资源收集日志数据用于分析和识别安全检查结果或可疑活动的根本原因。检测性行为图提供了可视化效果，可帮助您确定可能的安全问题的性质和范围，并开展有效的调查。 See Also https://aws.amazon.com/detective/.
Amazon DocumentDB (with MongoDB compatibility)	一项托管数据库服务，可用来在云中设置、操作和扩展与 MongoDB 兼容的数据库。 See Also https://aws.amazon.com/documentdb/.
Amazon DynamoDB	一种完全托管的 NoSQL 数据库服务，提供快速可预测的性能，同时还能够实现无缝扩展。 See Also https://aws.amazon.com/dynamodb/.
Amazon DynamoDB Encryption Client	一个软件库，可帮助您在将表数据发送到 Amazon DynamoDB (p. 70) 之前保护该数据。
适用于 Titan 的 Amazon DynamoDB 存储后端	一个基于 Amazon DynamoDB 实施的适用于 Titan 图数据库的存储后端。Titan 是针对存储和查询图形进行优化的可扩展图形数据库。 See Also https://aws.amazon.com/dynamodb/.
Amazon DynamoDB Streams	一项 Amazon 服务，可用于在任何 Amazon DynamoDB 表中捕获按时间排序的项目级修改序列。该服务还会将这类信息存储在日志中长达 24 个小时。应用程序可访问此日志，并在数据项目修改前后近乎实时地查看所显示的数据项目。 See Also https://aws.amazon.com/dynamodb/.
由 Amazon EBS-backed AMI	一种 Amazon Machine Image (AMI) (p. 72)，其实例使用 Amazon EBS (p. 70) volume (p. 116) 作为其根设备。将这种实例与从实例后端 (p. 95)启动的实例进行比较，后者使用 实例存储 (p. 95) 作为根设备。
Amazon EC2	一项 Web 服务，可用于启动和管理 Amazon 数据中心内的 Linux/UNIX 和 Windows Server 实例 (p. 95)。 See Also Amazon Elastic Compute Cloud (Amazon EC2), https://aws.amazon.com/ec2.
Amazon EC2 Auto Scaling	一项 Web 服务，可用于根据用户定义的策略 (p. 103)、时间安排和运行状况检查 (p. 93)自动启动或终止实例。 See Also https://aws.amazon.com/ec2/autoscaling.
Amazon Elastic Block Store (Amazon EBS)	一项 Web 服务，可用于提供数据块级存储卷 (p. 116)或与 EC2 实例 (p. 89)搭配使用。 See Also https://aws.amazon.com/ebs.
Amazon Elastic Compute Cloud (Amazon EC2)	一项 Web 服务，您可以使用该服务启动和管理 Amazon 数据中心内的 Linux/UNIX 和 Windows Server 实例 (p. 95)。 See Also https://aws.amazon.com/ec2.
Amazon Elastic Container Registry (Amazon ECR)	一个完全托管式 Docker 容器注册表，您可以使用该注册表存储、管理和部署 Docker 容器映像。Amazon ECR 与 Amazon Elastic Container Service (Amazon ECS) (p. 70) 和 Amazon Identity and Access Management (IAM) (p. 78) 集成。 See Also https://aws.amazon.com/ecr.
Amazon Elastic Container Service (Amazon ECS)	一项高度可扩展的快速 容器 (p. 85) 管理服务，您可以使用该服务运行、停止和管理 EC2 实例 集群 (p. 84) 上的 Docker 容器。 See Also https://aws.amazon.com/ecs.
Amazon Elastic File System (Amazon EFS)	适用于 EC2 (p. 70) 实例 (p. 95)的文件存储服务。Amazon EFS 提供可用于创建和配置文件系统的界面。Amazon EFS 存储容量会随着您添加和删除文件而自动增加和缩减。 See Also https://aws.amazon.com/efs/.

Amazon Elastic Kubernetes Service (Amazon EKS)	一项托管服务，您可以使用该服务在 Amazon 上运行 Kubernetes，无需支持或维护您自己的 Kubernetes 控制面板。 See Also https://aws.amazon.com/eks/.
Amazon Elastic Transcoder	一项基于云的媒体转码服务。Elastic Transcoder 是一个高度可扩展的工具，用于将媒体文件从其源格式转换（或转码）为将在智能手机、平板电脑和 PC 等设备上播放的版本。 See Also https://aws.amazon.com/elastictranscoder/.
Amazon ElastiCache	一个 Web 服务，可让您轻松地在云中部署、操作和扩展内存中缓存。该服务允许从快速的托管内存中缓存中检索信息，而无需完全依赖于速度较慢的基于磁盘的数据库，从而提高了 Web 应用程序的性能。 See Also https://aws.amazon.com/elasticache/.
Amazon OpenSearch Service (OpenSearch Service)	一项 Amazon 托管服务，可用于在 Amazon Web Services 云 中部署、操作和扩展 OpenSearch（一个开源搜索和分析引擎）。Amazon OpenSearch Service（OpenSearch Service）还提供安全性选项、高可用性、数据持久性以及对 OpenSearch API 的直接访问权限。 See Also https://aws.amazon.com/elasticsearch-service.
Amazon EMR	一项 Web 服务，您可以使用该服务高效地处理大量数据。Amazon EMR 使用 Hadoop (p. 93) 处理方法，并结合多个 Amazon 产品来执行以下各项任务：Web 索引、数据挖掘、日志文件分析、机器学习、科学模拟以及数据仓库。 See Also https://aws.amazon.com/elasticmapreduce.
Amazon EventBridge	一种无服务器事件总线服务，可用于将应用程序与来自各种来源的数据连接起来，并将这些数据路由到目标，例如 Amazon Lambda。您可以设置路由规则来确定发送数据的目的地，以便构建能够实时响应所有数据源的应用程序架构。 See Also https://aws.amazon.com/eventbridge/.
Amazon Forecast	一种完全托管式服务，可使用统计和机器学习算法生成高度精确的时间序列预测。 See Also https://aws.amazon.com/forecast/.
Amazon GameLift	一个托管服务，可用于部署、操作和扩展基于会话的多玩家游戏。 See Also https://aws.amazon.com/gamelift/.
Amazon GuardDuty	一项持续的安全监控服务。Amazon GuardDuty 可帮助识别 Amazon 环境中的意外活动和潜在的未经授权或恶意活动。 See Also https://aws.amazon.com/guardduty/.
Amazon Inspector	一项自动安全评估服务，有助于提高部署在 Amazon 上的应用程序的安全性与合规性。Amazon Inspector 会自动评估应用程序的漏洞以及偏离最佳实践的情况。执行评估后，Amazon Inspector 会生成一份详细的报告，其中包含按优先顺序排列的补救步骤。 See Also https://aws.amazon.com/inspector.
Amazon Kinesis	一个面向 Amazon 上的流数据的平台。Kinesis 提供可简化流数据的加载和分析的服务。 See Also https://aws.amazon.com/kinesis/.
Amazon Kinesis Data Firehose	一项完全托管式服务，用于将流数据加载到 Amazon 上。Kinesis Data Firehose 可以捕获并自动将流数据加载到 Amazon S3 (p. 74) 和 Amazon Redshift (p. 73) 中，使您能够利用现有的业务情报工具和控制面板执行接近实时的分析。Kinesis Data Firehose 可以弹性伸缩以满足数据吞吐量要求，并且无需进行日常管理。它还可以在加载数据之前对数据进行批处理、压缩和加密。 See Also https://aws.amazon.com/kinesis/firehose/.
Amazon Kinesis Data Streams	一项 Web 服务，可用于构建处理或分析流数据以满足具体需求的自定义应用程序。Amazon Kinesis Data Streams 每小时可从数十万个来源连续捕获和存储数 TB 的数据。

	See Also https://aws.amazon.com/kinesis/streams/.
Amazon Lightsail	您可以使用 Lightsail 通过 Amazon 启动和管理虚拟私有服务器。Lightsail 提供了捆绑套餐，其中包括部署虚拟专用服务器所需的一切，月度费用较低。 See Also https://aws.amazon.com/lightsail/.
Amazon Lookout for Equipment	一项机器学习服务，可使用安装在工厂设备上的传感器的数据检测异常行为，以便可以在机器发生故障之前采取措施。 See Also https://aws.amazon.com/lookout-for-equipment/.
Amazon Lookout for Vision	一种机器学习服务，可使用计算机视觉（CV）发现工业产品中的缺陷。Amazon Lookout for Vision 可以发现工业产品中缺失的组件、车辆或结构的损坏、生产线中的异常，甚至硅片或者质量至关重要的任何其他实物中的微小缺陷。 See Also https://aws.amazon.com/lookout-for-vision/.
Amazon Lumberyard	一个用于创建高品质游戏的跨平台 3D 游戏引擎。您可以将游戏与 Amazon Web Services 云 的计算和存储相关联，并吸引 Twitch 上爱好者的关注。 See Also https://aws.amazon.com/lumberyard/.
Amazon Machine Image（AMI）	一个加密计算机镜像，存储在 Amazon Elastic Block Store (Amazon EBS) (p. 70) 或 Amazon Simple Storage Service (p. 74) 中。AMI 的功能类似于计算机根驱动器的模板。它们包含操作系统，还可以包括软件和应用程序层，如数据库服务器、中间件和 Web 服务器。
Amazon Machine Learning	一个基于云的服务，该服务通过查找现有数据中的模式来创建机器学习 (ML) 模型，然后使用这些模型处理新数据并生成预测。 See Also http://aws.amazon.com/machine-learning/.
Amazon Macie	一项安全服务，可使用机器学习自动发现、分类和保护 Amazon 中的敏感数据。 See Also http://aws.amazon.com/macie/.
Amazon Managed Blockchain	一个完全托管的服务，用于使用流行的开源框架创建和管理可扩展的区块链网络。 See Also http://aws.amazon.com/managed-blockchain/.
Amazon Managed Grafana	一种安全的完全托管式数据可视化服务，您可以使用该服务即时查询、关联和可视化来自多个数据来源的运行指标、日志和跟踪。 See Also https://aws.amazon.com/grafana/.
Amazon Managed Service for Prometheus	一项服务，可用于为容器提供高度可用和安全的托管式监控。 See Also https://aws.amazon.com/prometheus/.
Amazon ML	See Amazon Machine Learning.
Amazon Mobile Analytics (Mobile Analytics)	一个用来大规模收集、可视化、理解并提取移动应用使用率数据的服务。 See Also https://aws.amazon.com/mobileanalytics.
Amazon Monitron	一个端到端系统，可使用机器学习（ML）检测工业机械中的异常行为。可使用 Amazon Monitron 实施预测性维护并减少计划外停机。 See Also https://aws.amazon.com/monitron/.
Amazon MQ	Apache ActiveMQ 的一项托管式消息代理服务，您可以使用该服务在云中设置和操作消息代理。 See Also https://aws.amazon.com/amazon-mq/.
Amazon Neptune	一种托管式图数据库服务，可用于构建并运行使用高度互连数据集的应用程序。Neptune 支持常见的图形查询语言 Apache TinkerPop Gremlin 和 W3C 的 SPARQL，可让您构建查询，高效地浏览高度互连数据集。 See Also https://aws.amazon.com/neptune/.
Amazon Personalize	一种人工智能服务，可用于创建个性化产品和内容推荐。

	See Also https://aws.amazon.com/personalize/.
Amazon Polly	一项文字转语音（TTS）服务，可将文字转换为听起来很自然的人类语音。Amazon Polly 提供各种语言的数十种逼真语音，因此您可以构建在许多不同国家/地区运行的支持语音的应用程序。 See Also https://aws.amazon.com/polly/.
Amazon QuickSight	一项云赋能的快速商业分析服务，您可以使用该服务实现可视化、执行分析以及快速地从数据中获得业务见解。 See Also https://aws.amazon.com/quicksight/.
Amazon Rekognition	一种机器学习服务，可识别图像或视频文件中的物体、人、文本、场景和活动（包括隐藏的内容）。借助 Amazon Rekognition Custom Labels，您可以创建自定义机器学习（ML）模型，以识别图像中特定于您业务的对象和场景。 See Also https://aws.amazon.com/rekognition/.
Amazon Redshift	一个完全托管的 PB 级云中数据仓库服务。借助 Amazon Redshift，您可使用现有的业务情报工具分析数据。 See Also https://aws.amazon.com/redshift/.
Amazon Relational Database Service (Amazon RDS)	一项 Web 服务，让用户能够在云中更轻松地设置、操作和扩展关系数据库。可为用户提供一个经济有效、容量可调的符合行业标准的关系数据库，并承担常见的数据库管理任务。 See Also https://aws.amazon.com/rds.
Amazon Resource Name (ARN)	一种引用 Amazon资源 (p. 106)（例如，arn:aws:iam::123456789012:user/division_abc/subdivision_xyz/Bob）的标准化方法。
Amazon Route 53	一项 Web 服务，您以使用该服务创建新 DNS 服务或将现有 DNS 服务迁移到云中。 See Also https://aws.amazon.com/route53.
Amazon S3	面向 Internet 的存储。您可以使用它随时从任何位置在 Web 上存储和检索任意数量的数据。 See Also Amazon Simple Storage Service (Amazon S3), https://aws.amazon.com/s3.
由 Amazon S3 支持的 AMI	See 实例存储支持的 AMI.
Simple Storage Service（Amazon S3）Glacier	一个安全、持久且成本较低的存储服务，适用于数据存档和长期备份。您可以可靠地存储大量或少量数据，且成本显著低于本地解决方案。S3 Glacier 已针对不频繁访问的数据进行优化（适合几小时检索一次的数据）。 See Also https://aws.amazon.com/glacier/.
Amazon Security Hub	一项服务，可以全面了解 Amazon 资源的安全状态。Security Hub 从 Amazon Web Services 账户 和服务中收集安全数据，并帮助您分析安全趋势，识别整个 Amazon 环境中的安全问题并确定其优先级。 See Also https://aws.amazon.com/security-hub/.
Amazon Silk	仅在 Fire OS 平板电脑和手机上可用的下一代 Web 浏览器。Amazon Silk 基于一个拆分架构（划分客户端和 Amazon Web Services 云 间的处理）构建，旨在创建更快、更具响应性的移动浏览体验。
Amazon Simple Email Service (Amazon SES)	一种针对应用程序的简单和经济高效的电子邮件解决方案。 See Also https://aws.amazon.com/ses.
Amazon Simple Notification Service (Amazon SNS)	一项 Web 服务，应用程序、用户和设备可通过该服务即时发送和接收云通知。 See Also https://aws.amazon.com/sns.
Amazon Simple Queue Service (Amazon SQS)	可靠的可扩展托管队列，用于存储计算机之间传输的消息。 See Also https://aws.amazon.com/sqs.

Amazon Simple Storage Service (Amazon S3)	面向 Internet 的存储。您可以使用它随时从任何位置在 Web 上存储和检索任意数量的数据。 See Also https://aws.amazon.com/s3.
Amazon Simple Workflow Service (Amazon SWF)	一种完全托管式服务，可帮助开发人员构建、运行和扩展具有并行或连续步骤的后台任务。Amazon SWF 的功能类似于 Amazon Web Services 云 中的状态跟踪器和任务协调器。 See Also https://aws.amazon.com/swf/.
Amazon Sumerian	一组用于在 Web 上创建并运行高质量 3D、增强现实 (AR) 和虚拟现实 (VR) 应用程序的工具。 See Also https://aws.amazon.com/sumerian/.
Amazon Textract	一项用于自动从扫描文档中提取文本和数据的服务。Amazon Textract 的功能远不止简单的光学字符识别（OCR），还可以识别表单中的字段内容以及存储在表中的信息。 See Also https://aws.amazon.com/textract/.
Amazon Transcribe	一种机器学习服务，可使用自动语音识别（ASR）快速准确地将语音转换为文本。 See Also https://aws.amazon.com/transcribe/.
Amazon Transcribe Medical	一种自动语音识别（ASR）服务，用于将医疗语音转文本功能添加到支持语音的临床文档应用程序中。 See Also https://aws.amazon.com/transcribe/medical/.
Amazon Translate	一种神经网络机器翻译服务，可提供快速、高质量和价格实惠的语言翻译。 See Also https://aws.amazon.com/translate/.
Amazon Virtual Private Cloud (Amazon VPC)	一项 Web 服务，用于预置您定义的 Amazon Web Services 云 虚拟网络的逻辑分隔部分。您可以选择自有的 IP 地址范围，创建子网 (p. 112)，以及配置路由表 (p. 107)和网络网关，从而控制您的虚拟联网环境。 See Also https://aws.amazon.com/vpc.
Amazon VPC	See Amazon Virtual Private Cloud (Amazon VPC).
Amazon Web Services (Amazon)	一个面向各种规模的公司的云中基础设施 Web 服务平台。 See Also https://aws.amazon.com/what-is-cloud-computing/.
Amazon WorkDocs	一个托管的安全企业文档存储和共享服务，具有管理控制和反馈功能。 See Also https://aws.amazon.com/workdocs/.
Amazon WorkLink	一个基于云的服务，可以在 iOS 手机中提供对内部网站和 Web 应用程序的安全访问。 See Also https://aws.amazon.com/worklink/.
Amazon WorkMail	一个托管的安全企业电子邮件和日历服务，可支持现有桌面和移动电子邮件客户端。 See Also https://aws.amazon.com/workmail/.
Amazon WorkSpaces	一项安全的托管式桌面计算服务，可用于预置基于云的桌面并向用户提供对来自受支持设备的文档、应用程序和资源 (p. 106)的访问权限。 See Also https://aws.amazon.com/workspaces/.
Amazon WorkSpaces Application Manager (Amazon WAM)	一项 Web 服务，用于部署和管理 WorkSpaces 的应用程序。Amazon WAM 通过将 Windows 桌面应用程序打包到虚拟化的应用程序容器中，加快软件部署、升级、修补和停用。 See Also https://aws.amazon.com/workspaces/applicationmanager.
AMI	See Amazon Machine Image（AMI）.
分析方案	Amazon CloudSearch (p. 69)：特定于语言的文本分析选项，适用于文本字段，用于控制词干分解和配置非索引字和同义词。

application	Amazon Elastic Beanstalk (p. 77)：组件的逻辑集合，包括环境、版本和环境配置。应用程序在概念上类似于文件夹。 Amazon CodeDeploy (p. 77)：唯一标识要部署的应用程序的名称。AmazonCodeDeploy 使用此名称以确保修订、部署配置和部署组的正确组合，并确保在部署期间引用部署组。
Application Auto Scaling	一项 Web 服务，可用于为 Amazon EC2 以外的 Amazon 资源（如 Amazon ECS 服务、Amazon EMR 集群和 DynamoDB 表）配置自动扩展。 See Also https://aws.amazon.com/autoscaling/.
应用程序账单	您的客户管理他们所购买的 Amazon DevPay 产品的位置。Web 地址为 http://www.amazon.com/dp-applications。
应用程序修订	Amazon CodeDeploy (p. 77)：一个包含源内容（例如，源代码、网页、可执行文件和部署脚本）的存档文件以及一个应用程序规范文件 (p. 75)。修订存储在 Amazon S3 (p. 74) 存储桶 (p. 83) 或 GitHub (p. 93) 存储库中。对于 Amazon S3，修订由其 Amazon S3 对象键以及其 ETag 和/或版本唯一标识。对于 GitHub，修订由其提交 ID 唯一标识。
应用程序规范文件	Amazon CodeDeploy (p. 77)：一个 YAML 格式的文件，用于将应用程序修订中的源文件映射到实例上的目的地。该文件还可用于指定已部署文件的自定义权限，以及指定在部署过程的各个阶段要在每个实例上运行的脚本。
应用程序版本	Amazon Elastic Beanstalk (p. 77)：应用程序的特定标记迭代，代表功能上一致的可部署的应用程序代码集。一个版本指向一个 Amazon S3 (p. 74) 对象（JAVA WAR 文件），其中包含应用程序代码。
AppSpec 文件	See 应用程序规范文件.
ARN	See Amazon Resource Name (ARN).
项目	Amazon CodePipeline (p. 77)：由管道操作处理的文件或更改的副本。
非对称加密	一种同时使用公钥和私钥的加密 (p. 90)方式。
异步退回邮件	一种 退回邮件 (p. 83)，如果 接收方 (p. 105) 最初接受电子邮件进行递送，后来未能递送，则会发生这种情况。
原子计数器	DynamoDB：递增或递减现有属性的值而不干扰其他写入请求的方法。
属性	一个基础数据元素，无需进一步分解。在 DynamoDB 中，属性在很多方面都类似于其他数据库系统中的字段或列。 Amazon Machine Learning：数据集的观察中唯一的指定属性。在表格数据（例如，电子表格或逗号分隔的值 (.csv) 文件）中，列标题代表属性，而行包含每个属性的值。
AUC	曲线下方的区域。一个行业标准指标，用于评估二进制分类机器学习模型的质量。AUC 测量模型的能力以预测更高的分数，对于正面示例，它们是“正确的”，对于负面示例，它们是“错误的”。AUC 指标返回从 0 到 1 的数值。接近 1 的 AUC 值指示高度准确的机器学习（ML）模型。
Aurora	See Amazon Aurora.
经过身份验证的加密	一种加密 (p. 90)方式，此加密提供已加密数据的机密性、数据完整性和身份验证保证。
身份验证	向系统证明身份的过程。
自动扩缩组	表示多个具有相似特征且被视为逻辑组以便进行实例扩展和管理的 EC2 实例 (p. 89)。

可用区：	一个 区域 (p. 106) 中的不同位置，用于与其他可用区的故障隔离，并向同一区域中的其他可用区提供低成本、低延迟的网络连接。
Amazon Application Discovery Service	一个 Web 服务，可帮助您通过标识数据中心内的 IT 资产 (包括服务器、虚拟机、应用程序、应用程序依赖项和网络基础设施) 来规划到 Amazon 的迁移。 See Also https://aws.amazon.com/about-aws/whats-new/2016/04/aws-application-discovery-service/.
Amazon AppSync	一个完全托管的企业级 GraphQL 服务，具有实时数据同步和离线编程功能。 See Also https://aws.amazon.com/appsync/.
Amazon Auto Scaling	一项完全托管式服务，可用于快速发现作为您的应用程序的一部分的可扩展 Amazon 资源并配置动态扩展。 See Also https://aws.amazon.com/autoscaling/.
Amazon Backup	一项托管式备份服务，可用于在云中以及本地集中管理和自动执行跨 Amazon 服务的数据备份。 See Also https://aws.amazon.com/backup/.
Amazon Billing and Cost Management	Amazon Web Services 云 计算模式，可让您按所需服务付费，并可根据需要确定用量。虽然资源 (p. 106)在您的账户下是动态的，但您需要支付分配这些资源所产生的成本。您还需要支付与这些资源关联的任何附带使用费（如数据传输或分配的存储）。 See Also https://aws.amazon.com/billing/new-user-faqs/.
Amazon Blockchain Templates	一项服务，可在 Amazon 上创建和部署开源区块链框架，例如 Ethereum 和 Hyperledger Fabric。 See Also https://aws.amazon.com/blockchain/templates/.
Amazon Certificate Manager (ACM)	一项 Web 服务，用于预置、管理和部署安全套接字层/传输层安全性 (p. 114) (SSL/TLS) 证书以用于 Amazon 服务。 See Also https://aws.amazon.com/certificate-manager/.
Amazon Private Certificate Authority (ACM PCA)	一个托管式私有证书颁发机构服务，可用于签发和撤销私有数字证书 (p. 84)。 See Also https://aws.amazon.com/certificate-manager/private-certificate-authority/.
Amazon Cloud Development Kit (Amazon CDK)	一个开源软件开发框架，可用于在代码中定义您的云基础设施并通过 Amazon CloudFormation 进行配置。 See Also https://aws.amazon.com/cdk/.
Amazon Cloud Map	一项服务，可用于创建和维护您的应用程序所依赖的后端服务和资源的映射。使用 Amazon Cloud Map，可以指定和发现 Amazon Web Services 云 资源。 See Also https://aws.amazon.com/cloud-map.
Amazon Cloud9	一种云端集成开发环境 (IDE)，用于编写、运行和调试代码。 See Also https://aws.amazon.com/cloud9/.
Amazon CloudFormation	一项服务，可用于编写或更改将相关 Amazon 资源 (p. 106) 作为一个整体进行创建和删除的模板。 See Also https://aws.amazon.com/cloudformation.
Amazon CloudHSM	一项 Web 服务，可通过在 Amazon Web Services 云 中使用专用的硬件安全模块（HSM）设备，帮助您满足数据安全方面的企业、合同和监管合规性的要求。 See Also https://aws.amazon.com/cloudhsm/.
Amazon CloudTrail	一个 Web 服务，可记录账户的 Amazon API 调用，并向您发送日志文件。记录的信息包括 API 调用者的身份、API 调用的时间、API 调用者的源 IP 地址、请求参数以及 Amazon 服务返回的响应元素。 See Also https://aws.amazon.com/cloudtrail/.

Amazon CodeBuild	一项完全托管的连续集成服务，可编译源代码、运行测试以及生成可供部署的软件包。 See Also https://aws.amazon.com/codebuild.
Amazon CodeCommit	一项完全托管式源控制服务，公司可以使用该服务托管安全且高度可扩展的私有 Git 存储库。 See Also https://aws.amazon.com/codecommit.
Amazon CodeDeploy	一项可以将代码自动部署到任意实例的服务，这些实例包括 EC2 实例 (p. 89) 和在本地运行的实例 (p. 95)。 See Also https://aws.amazon.com/codedeploy.
Amazon CodeDeploy 代理	一个软件包，在某个实例上安装和配置此软件包后，将支持在 CodeDeploy 部署中使用该实例。
Amazon CodePipeline	一项可实现快速可靠的应用程序更新的持续交付服务。 See Also https://aws.amazon.com/codepipeline.
Amazon Command Line Interface (Amazon CLI)	一个用于管理 Amazon 服务的可下载且可配置的统一工具。从命令行控制多个 Amazon 服务并通过脚本自动执行这些服务。 See Also https://aws.amazon.com/cli/.
Amazon Config	一项完全托管的服务，该服务提供了 Amazon 资源 (p. 106) 清单、配置历史记录和配置更改通知以改善安全性和管理。您可以创建规则来自动检查 Amazon Config 记录的 Amazon 资源的配置。 See Also https://aws.amazon.com/config/.
Amazon Database Migration Service	一个 Web 服务，可帮助您在许多广泛使用的商用数据库和开源数据库中迁入或迁出数据。 See Also https://aws.amazon.com/dms.
Amazon Data Pipeline	一项 Web 服务，用于在指定的时间间隔内，在不同的 Amazon 计算和存储服务以及本地数据源之间处理和移动数据。 See Also https://aws.amazon.com/datapipeline.
Amazon Device Farm (Device Farm)	一个应用程序测试服务，可让开发人员在由 Amazon 托管的真实、实际的手机和平板电脑上测试您的 Android、iOS 和 Fire OS 设备。 See Also https://aws.amazon.com/device-farm.
Amazon Direct Connect	一个 Web 服务，可让您轻松建立从本地到 Amazon 的专用网络连接。借助 Amazon Direct Connect，您可以建立 Amazon 与数据中心、办公室或主机托管环境之间的私有连接。 See Also https://aws.amazon.com/directconnect.
Amazon Directory Service	一项托管式服务，用于将 Amazon 资源 (p. 106)连接到现有的本地 Microsoft Active Directory 或在 Amazon Web Services 云 中设置和操作新的独立目录。 See Also https://aws.amazon.com/directoryservice.
Amazon Elastic Beanstalk	一项 Web 服务，用于在 Amazon Web Services 云 中部署和管理应用程序，而无需为运行这些应用程序的基础设施操心。 See Also https://aws.amazon.com/elasticbeanstalk.
AWS Elemental MediaConnect	一项服务，广播者和其他高级视频提供商可用于可靠地将实时视频摄取到 Amazon Web Services 云 中，并将其分发到 Amazon Web Services 云 内部或外部的多个目标。 See Also https://aws.amazon.com/mediaconnect.
AWS Elemental MediaConvert	一项基于文件的视频转换服务，可将媒体转换为传统广播和到多屏设备的 Internet 流媒体传输所需的格式。 See Also https://aws.amazon.com/mediaconvert.

AWS Elemental MediaLive	一项视频服务，可用于创建适合广播和流媒体传输的实时输出。 See Also https://aws.amazon.com/medialive.
AWS Elemental MediaPackage	一项适时打包和来源服务，可用于规定适合各种设备的高度安全可靠的实时输出格式。 See Also https://aws.amazon.com/mediapackage.
AWS Elemental MediaStore	一项专为媒体优化的存储服务，提供大规模交付实时和按需视频内容所需的性能、一致性和低延迟。 See Also https://aws.amazon.com/mediastore.
AWS Elemental MediaTailor	一项视频服务，可用于向观看者提供有针对性的广告，同时在 OTT 视频应用程序中保持广播质量。 See Also https://aws.amazon.com/mediatailor.
Amazon Encryption SDK	一个客户端加密库，您可以使用该库通过行业标准和最佳实践来加密和解密数据。 See Also https://aws.amazon.com/blogs/security/tag/aws-encryption-sdk/.
Amazon Firewall Manager	一项服务，与 Amazon WAF 结合使用可简化跨多个账户和多种资源的 Amazon WAF 管理和维护任务。利用 Amazon Firewall Manager，您只需设置您的防火墙规则一次。该服务会跨您的账户和资源自动应用规则，即使您添加了新资源。 See Also https://aws.amazon.com/firewall-manager.
Amazon Global Accelerator	一项用于创建加速器的网络层服务，可将流量定向到 Amazon 全局网络上的最佳终端节点。这可提高全球受众使用的 Internet 应用程序的可用性和性能。 See Also https://aws.amazon.com/global-accelerator.
Amazon Glue	完全托管的提取、转换和加载 (ETL) (p. 92)服务，您可用于对数据编目录并加载这些数据以进行分析。借助 Amazon Glue，您可以发现您的数据、开发脚本以将源转换为目标并安排在无服务器环境中运行 ETL 任务。 See Also https://aws.amazon.com/glue.
Amazon GovCloud (US)	一个独立的 Amazon Web Services 区域，可用于托管云中的敏感工作负载，并确保此工作符合美国政府的法规和合规性要求。Amazon GovCloud (US)区域符合美国的国际武器贸易条例 (ITAR)、联邦风险与授权管理项目 (FedRAMP) 要求、国防部 (DOD) 云安全要求指南 (SRG) 级别 2 和 4 以及刑事司法信息服务 (CJIS) 安全策略要求。 See Also https://aws.amazon.com/govcloud-us/.
Amazon IAM Identity Center (successor to Amazon Single Sign-On)	一种基于云的服务，可集中管理用户及其对 Amazon Web Services 账户 和云应用程序的访问。您可以控制 Amazon Organizations 中所有 Amazon Web Services 账户的单点登录访问权限和用户权限。 See Also https://aws.amazon.com/single-sign-on/.
Amazon Identity and Access Management (IAM)	一项 Web 服务，Amazon Web Services (Amazon) (p. 74) 客户可用于在 Amazon 中管理用户和用户权限。 See Also https://aws.amazon.com/iam.
Amazon Import/Export	一项服务，用于在 Amazon 和便携式存储设备之间传输大量数据。 See Also https://aws.amazon.com/importexport.
Amazon IoT 核心	一个托管的云平台，使互联设备可以轻松安全地与云应用程序及其他设备交互。 See Also https://aws.amazon.com/iot.
AmazonIoT 1-Click	一项服务，简单设备可用于启动 Amazon Lambda 函数。 See Also https://aws.amazon.com/iot-1-click.
Amazon IoT Analytics	一项完全托管的服务，可用于对大量 IoT 数据运行复杂的分析。 See Also https://aws.amazon.com/iot-analytics.

Amazon IoT Device Defender	一项 Amazon IoT 安全服务，可用于审计设备的配置，监控互联设备以检测异常行为，并降低安全风险。 See Also https://aws.amazon.com/iot-device-defender.
Amazon IoT Device Management	一项用于大规模地安全搭载、组织、监控和远程管理 IoT 设备的服务。 See Also https://aws.amazon.com/iot-device-management.
Amazon IoT Events	一项完全托管式 Amazon IoT 服务，您可以使用该服务检测来自 IoT 传感器和应用程序的事件并做出响应。 See Also https://aws.amazon.com/iot-events.
Amazon IoT Greengrass	一种软件，可用于以安全的方式为互联设备运行本地计算、消息收发、数据缓存、同步和机器学习（ML）推理功能。 See Also https://aws.amazon.com/greengrass.
Amazon IoT SiteWise	一种托管式服务，可用于从工业设备中大规模收集、组织和分析数据。 See Also https://aws.amazon.com/iot-sitewise.
Amazon IoT Things Graph	一项服务，实现不同设备和 Web 服务间的可视化连接，以构建 IoT 应用程序。 See Also https://aws.amazon.com/iot-things-graph.
Amazon Key Management Service (Amazon KMS)	一个托管服务，可让您轻松创建和控制用于加密数据的加密 (p. 90)密钥。 See Also https://aws.amazon.com/kms.
Amazon Lambda	一项 Web 服务，可用于运行代码，而无需预置或管理服务器。您几乎可以为任何类型的应用程序或后端服务运行代码，而无需管理。您可以将您的代码设置为自动从其他 Amazon 服务启动，或者直接从任何 Web 或移动应用程序调用。 See Also https://aws.amazon.com/lambda/.
Amazon 托管式密钥	Amazon Key Management Service (Amazon KMS) (p. 79) 中的一种 KMS 密钥。
Amazon 托管式策略	Amazon 创建和管理的 IAM (p. 78) 管理的策略 (p. 98)。
Amazon Web Services Management Console	一个图形界面，可用于管理计算、存储和其他云资源 (p. 106)。 See Also https://aws.amazon.com/console.
Amazon Management Portal for vCenter	一项 Web 服务，可用于通过 VMware vCenter 管理您的 Amazon 资源 (p. 106)。您可以将门户作为 vCenter 插件安装在您现有的 vCenter 环境中。安装后，您可以将 VMware 虚拟机迁移到 Amazon EC2 (p. 70)，并从 vCenter 内部管理 Amazon 资源。 See Also https://aws.amazon.com/ec2/vcenter-portal/.
Amazon Marketplace	一个 Web 门户，合乎要求的合作伙伴可在其中向 Amazon 客户营销和出售其软件。AmazonMarketplace 是一个在线软件商店，可帮助客户寻找、购买和快速启动在 Amazon 上运行的软件和服务。 See Also https://aws.amazon.com/partners/aws-marketplace/.
Amazon Migration Hub	一项服务，可用于提供单个位置以跟踪跨多个 Amazon 工具和合作伙伴解决方案的迁移任务。 See Also https://aws.amazon.com/migration-hub/.
Amazon Mobile Hub (Mobile Hub)	一个用于构建、测试和监视移动应用程序的集成控制台。 See Also https://aws.amazon.com/mobile.
Amazon Mobile SDK	一个软件开发工具包，其中包含的库、代码示例和文档，可帮助您构建面向 iOS、Android、Fire OS、Unity 和 Xamarin 平台的高质量移动应用程序。 See Also https://aws.amazon.com/mobile/sdk.
Amazon OpsWorks	一个配置管理服务，可帮助您使用 Chef 配置和操作实例和应用程序组。您可以定义应用程序的架构和每个组件的规范，包括软件包安装、软件配置和存储等资源 (p. 106)。您可以根据时间、负载或生命周期事件等来自动执行任务。

	See Also https://aws.amazon.com/opsworks/.
Amazon Organizations	一项账户管理服务，可用于将多个 Amazon Web Services 账户 整合到您创建并集中管理的组织中。 See Also https://aws.amazon.com/organizations/.
Amazon Resource Access Manager	一项服务，可用来与 Amazon Organizations 中的任何 Amazon Web Services 账户或组织共享资源。 See Also https://aws.amazon.com/ram/.
Amazon ParallelCluster	一个 Amazon 支持的开源集群管理工具，可帮助您在 Amazon Web Services 云 中部署和管理高性能计算 (HPC) 集群。
适用于 C++ 的 Amazon SDK	一个开发工具包，可提供面向许多 Amazon 服务（包括 Amazon S3 (p. 74)、Amazon EC2 (p. 70)、Amazon DynamoDB (p. 70) 等）的 C++ API。这个可下载的单一软件包中包含 Amazon C++ 库、代码示例和文档。 See Also https://aws.amazon.com/sdk-for-cpp/.
适用于 Go 的 Amazon SDK	一个开发工具包，用于将 Go 应用程序与整个 Amazon 服务套件集成。 See Also https://aws.amazon.com/sdk-for-go/.
Amazon SDK for Java	一个开发工具包，可提供面向许多 Amazon 服务（包括 Amazon S3 (p. 74)、Amazon EC2 (p. 70)、Amazon DynamoDB (p. 70) 等）的 Java API 操作。这个可下载的单一软件包中包含 Amazon Java 库、代码示例和文档。 See Also https://aws.amazon.com/sdk-for-java/.
浏览器中的 Amazon SDK for JavaScript	一个开发工具包，用于从浏览器中运行的 JavaScript 代码访问 Amazon 服务。使用 Web 联合身份验证通过 Facebook、Google 或 Login with Amazon 来对用户进行身份验证。将应用程序数据存储在 Amazon DynamoDB (p. 70) 中，并将用户文件保存到 Amazon S3 (p. 74)。 See Also https://docs.aws.amazon.com/sdk-for-javascript/v2/developer-guide/.
适用于 Node.js 中的 JavaScript 的 Amazon SDK	一个开发工具包，用于从 Node.js 中的 JavaScript 访问 Amazon 服务。此 SDK 提供面向 Amazon 服务（包括 Amazon S3 (p. 74)、Amazon EC2 (p. 70)、Amazon DynamoDB (p. 70) 和 Amazon Simple Workflow Service (Amazon SWF) (p. 74)）的 JavaScript 对象。这个可下载的单一数据包中包含 Amazon JavaScript 库和文档。 See Also https://docs.aws.amazon.com/sdk-for-javascript/v2/developer-guide/.
Amazon SDK for .NET	一个开发工具包，提供面向 Amazon 服务（包括 Amazon S3 (p. 74)、Amazon EC2 (p. 70)、IAM (p. 78) 等）的 .NET API 操作。可以将此软件开发工具包作为 NuGet 上多个特定于服务的程序包下载。 See Also https://aws.amazon.com/sdk-for-net/.
Amazon SDK for PHP	一个开发工具包和开源 PHP 库，用于将您的 PHP 应用程序与 Amazon 服务（如 Amazon S3 (p. 74)、Simple Storage Service（Amazon S3）Glacier (p. 73) 和 Amazon DynamoDB (p. 70)）集成。 See Also https://aws.amazon.com/sdk-for-php/.
Amazon SDK for Python (Boto)	一个开发工具包，用于使用 Python 访问 Amazon 服务，例如 Amazon EC2 (p. 70)、Amazon EMR (p. 71)、Amazon EC2 Auto Scaling (p. 70)、Amazon Kinesis (p. 71) 或 Amazon Lambda (p. 79)。 See Also http://boto.readthedocs.org/en/latest/.
Amazon SDK for Ruby	一个用于从 Ruby 访问 Amazon 服务的开发工具包。此 SDK 提供面向许多 Amazon 服务（包括 Amazon S3 (p. 74)、Amazon EC2 (p. 70)、Amazon DynamoDB (p. 70) 等）的 Ruby 类。这个可下载软件包中包含 Amazon Ruby 库和文档。

	See Also https://aws.amazon.com/sdk-for-ruby/.
Amazon Secrets Manager	一项服务，用于安全加密、存储和轮换数据库和其他服务的凭证。 See Also https://aws.amazon.com/secrets-manager/.
Amazon Security Token Service (Amazon STS)	一个 Web 服务，用于为 Amazon Identity and Access Management (IAM) (p. 78) 用户或进行身份验证的用户（联合身份用户 (p. 92)）请求具有有限权限的临时凭证。 See Also https://aws.amazon.com/iam/.
Amazon Service Catalog	一项 Web 服务，可帮助组织创建和管理已批准在 Amazon 上使用的 IT 服务的目录。这些 IT 服务可以包含所有内容，从虚拟机映像、服务器、软件和数据库到完整的多层次应用程序架构一应俱全。 See Also https://aws.amazon.com/servicecatalog/.
Amazon Shield	一种服务，有助于保护您的资源（例如 Amazon EC2 实例、Elastic Load Balancing 负载均衡器、Amazon CloudFront 分配和 Route 53 托管区域）免受 DDoS 攻击。Amazon Shield 自动包含在内，除了已为 Amazon WAF 和其他 Amazon 服务支付的费用外，无任何附加成本。为了针对 DDoS 攻击获得附加保护，Amazon 提供了 Amazon Shield Advanced。 See Also https://aws.amazon.com/shield.
Amazon Step Functions	一项 Web 服务，可以将分布式应用程序的各组件作为可视化工作流中的一系列步骤进行协调。 See Also https://aws.amazon.com/step-functions/.
Amazon Snowball	一种 PB 规模数据传输解决方案，使用安全的设备将大量数据移入或移出 Amazon Web Services 云。 See Also https://aws.amazon.com/snowball.
Storage Gateway	一项 Web 服务，可用于将本地软件设备与基于云的存储连接起来。Storage Gateway 提供组织的本地 IT 环境和 Amazon 存储基础设施之间的无缝且安全的集成。 See Also https://aws.amazon.com/storagegateway/.
Amazon Toolkit for Eclipse	一个适用于 Eclipse Java 集成开发环境（IDE）的开源插件，有助于更轻松地使用 Amazon Web Services 开发、调试和部署 Java 应用程序。 See Also https://aws.amazon.com/eclipse/.
Amazon Toolkit for JetBrains	Jetbrains 研发的面向集成开发环境（IDE）的开源插件，有助于更轻松地使用 Amazon Web Services 开发、调试和部署无服务器应用程序。 See Also https://aws.amazon.com/intellij/, https://aws.amazon.com/pycharm/.
Amazon Toolkit for Visual Studio	Visual Studio 的扩展，有助于使用 Amazon Web Services 开发、调试和部署 .NET 应用程序。 See Also https://aws.amazon.com/visualstudio/.
Amazon Toolkit for Visual Studio Code	适用于 Visual Studio Code (VS Code) 编辑器的开源插件，有助于更轻松地使用 Amazon Web Services 开发、调试和部署应用程序。 See Also https://aws.amazon.com/visualstudiocode/.
Amazon Tools for PowerShell	一组 PowerShell cmdlet，可帮助开发人员和管理员从 PowerShell 脚本环境管理其 Amazon 服务。 See Also https://aws.amazon.com/powershell/.
Amazon Toolkit for Microsoft Azure DevOps	提供您可在 VSTS 的生成和发布定义中用来与 Amazon 服务交互的任务。 See Also https://aws.amazon.com/vsts/.
Amazon Trusted Advisor	一个 Web 服务，可检查您的 Amazon 环境，并提供有关节省资金、提高系统可用性和性能以及帮助修补安全漏洞的建议。

	See Also https://aws.amazon.com/premiumsupport/trustedadvisor/.
Amazon VPN CloudHub	使用简单的星型拓扑连接模型（无论有没有 VPC (p. 116)）来支持在分支机构之间建立安全通信。
Amazon WAF	一个 Web 应用程序防火墙服务，可通过根据您指定的条件来允许或阻止 Web 请求，从而控制对内容的访问。例如，您可以根据标头值或请求源自的 IP 地址来筛选访问。Amazon WAF 帮助保护 Web 应用程序免遭常见 Web 漏洞的攻击，这些漏洞会影响应用程序可用性、降低安全性或占用过多资源。 See Also https://aws.amazon.com/waf/.
Amazon X-Ray	一项 Web 服务，用于收集有关应用程序所处理请求的数据。X-Ray 提供用于查看、筛选和获取数据洞察力的工具，以发现问题和优化机会。 See Also https://aws.amazon.com/xray/.

B，

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) | O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

基本监控	监控每 5 分钟产生一次的 Amazon 提供的指标。
批处理	See 文档批处理.
BGP ASN	边界网关协议自治系统编号。网络的唯一标识符，用于 BGP 路由。Amazon EC2 (p. 70) 支持 1 到 65335 范围内的所有双字节 ASN 编号，但 7224 除外，该编号是预留编号。
批量预测	Amazon Machine Learning：一个一次性（异步）处理多个输入数据观察的操作。与实时预测不同，批量预测在所有预测处理完之前不可用。 See Also 实时预测.
计费	See Amazon Billing and Cost Management.
二进制属性	Amazon Machine Learning：一个属性，可能具有两个可能的值之一。有效的正值为 1、y、yes、t 和 true。有效的负值为 0、n、no、f 和 false。Amazon Machine Learning 输出 1 表示正值，0 表示负值。 See Also 属性.
二进制分类模型	Amazon Machine Learning：一个可预测问题答案的机器学习模型，其中可以二进制变量形式表示答案。例如，答案为“1”或“0”、“yes”或“no”、“will click”或“will not click”的问题是具有二进制答案的问题。二进制分类模型的结果始终为“1”（对于“true”或肯定答案）或“0”（对于“false”或否定答案）。
数据块	一个数据集。Amazon EMR (p. 71) 将大量数据细分成子集。每个子集称为一个数据块。Amazon EMR 为每个数据块分配一个 ID，并使用哈希表跟踪数据块处理。
块储存设备	支持在固定大小的数据块、扇区或群集中读取和（可选）写入数据的一种存储设备。
块设备映射	每个 AMI (p. 72) 和实例 (p. 95)的映射结构，它指定与实例相连接的块设备。
蓝/绿部署	CodeDeploy：一种部署方法，在此方法中，部署组中的实例（原始环境）由一组不同的实例（替换环境）替换。
引导操作	用户指定的默认或自定义操作，它在 Hadoop (p. 93) 启动前在任务流程的所有节点上运行脚本或应用程序。

边界网关协议自治系统编号	See BGP ASN.
退回邮件	一次失败的电子邮件递送尝试。
违例	Amazon EC2 Auto Scaling (p. 70)：超出用户设置的阈值（上限或下限）的情况。如果违例持续时间很长（如违例持续时间参数所设置），则它可能启动 扩展活动 (p. 108)。
桶	Amazon Simple Storage Service (Amazon S3) (p. 74)：用于存储对象的容器。每个对象都储存在一个存储桶中。举例来说，如果名为 `photos/puppy.jpg` 的对象存储在 *`DOC-EXAMPLE-BUCKET`* 存储桶中，则授权用户可以通过 URL `https://`*`s3-bucket-endpoint`*`/`*`DOC-EXAMPLE-BUCKET`*`/photos/puppy.jpg` 访问该对象。
存储桶拥有者	在 Amazon S3 (p. 74) 中拥有桶 (p. 83)的个人或组织。就像 Amazon 是域名 Amazon.com 的唯一拥有者一样，一个存储桶只能归一个用户或组织所有。
捆绑	用于创建Amazon Machine Image（AMI） (p. 72) 的常用术语。其特指创建实例存储支持的 AMI (p. 95)。

C

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) | O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

缓存集群	在多个缓存节点 (p. 83)上分配的逻辑缓存。缓存集群可设置为具有特定数量的缓存节点。
缓存群集标识符	客户提供的缓存集群标识符，此标识符必须为 Amazon Web Services 区域 (p. 106) 中该客户的唯一标识符。
缓存引擎版本	在缓存节点上运行的 Memcached 服务的版本。
缓存节点	固定大小、网络挂载的安全 RAM 区块。每个缓存节点都运行一个 Memcached 软件实例，有其自己的 DNS 名称和端口。支持多种缓存节点类型，每种可有不同的关联内存量。
缓存节点类型	用于运行缓存节点的 EC2 实例 (p. 89)类型。
缓存参数组	可应用于一个或多个缓存集群的缓存引擎参数值的容器。
缓存安全组	一个由 ElastiCache 维护的组，可用于将入站授权合并到主机的缓存节点，该主机属于通过控制台、API 或命令行工具指定的 Amazon EC2 (p. 70) 安全组 (p. 109)。
活动	Amazon Personalize (p. 72)：部署的解决方案版本（经过训练的模型），具有预置的专用事务容量，用于为您的应用程序用户创建实时建议。创建活动后，您可以使用 `getRecommendations` 或者 `getPersonalizedRanking` 个性化操作获取建议。 See Also 推荐, 解决方案版本.
标准访问策略	一个您可以应用于桶 (p. 83)或对象的标准访问控制策略。其选项包括：私有读取、公有读取、公有读写、经身份验证的读取。
标准化	将数据转换为某种服务（如 Amazon S3 (p. 74)）可识别的标准格式的过程。
capacity	某个给定时间的可用计算规模量。每个 自动扩缩组 (p. 75) 都定义有最小和最大计算规模。扩展活动 (p. 108)会在定义的最小值和最大值内增加或减少容量。

笛卡尔积处理器	一种计算笛卡尔积的处理器。也称作笛卡尔数据处理器。
笛卡尔积	从多个集返回积的数学运算。
CDN	See 内容分发网络 (CDN).
证书	一种凭证，供某些 Amazon 产品用来对 Amazon Web Services 账户 (p. 68) 和用户进行身份验证。也称作 X.509 证书 (p. 116)。该凭证与私有密钥成对使用。
应计费的资源	使用时会产生费用的功能或服务。有些 Amazon 产品免费，但也有一些产品收费。例如，在 Amazon CloudFormation (p. 76) 堆栈 (p. 111) 中，已创建的 Amazon 资源 (p. 106)会产生费用。具体费用取决于使用负载。在创建实例、堆栈或其他资源之前，可使用 处的 Amazon Web Services 简单月度成本结算器估计您的费用。
CIDR 块	无类域间路由。一种 Internet 协议地址分配和路由聚合方法。 See Also 无类域间路由 on Wikipedia.
密文	已加密的 (p. 90)信息，与明文 (p. 103)（它是未加密的信息）相对。
分类	机器学习中的一种问题，此问题寻求将数据示例放置 (分类) 到一个种类或“类别”中。通常，会对分类问题建模以便从两个种类（类别）中选择一个种类（类别）。这些问题是二进制分类问题。有两种以上的种类（类别）的问题被称作“多类别分类”问题。 See Also 二进制分类模型, 多类别分类模型.
CLI	See Amazon Command Line Interface (Amazon CLI).
Cloud Directory	See Amazon Cloud Directory (Cloud Directory).
云服务提供商（CSP）	向订阅者提供对 Internet 上托管的计算、存储和软件服务的访问的公司。
CloudHub	See Amazon VPN CloudHub.
集群	容器实例 (p. 85)的逻辑分组，您可以将任务 (p. 113)放置在其中。 Amazon OpenSearch Service (OpenSearch Service) (p. 71)：运行 Amazon OpenSearch Service（OpenSearch Service）并操作您的 OpenSearch Service 域所需的一个或多个数据节点、可选专用主节点和存储的逻辑分组。 See Also 数据节点, 专用主节点, node.
群集计算实例	一种 实例 (p. 95)，提供强大的 CPU 以及增强的网络连接性能，非常适合高性能计算 (HPC) 应用程序和其他高要求的网络绑定型应用程序。
群集置放群组	一个在实例 (p. 95)之间提供更低延迟和高带宽连接的逻辑 群集计算实例 (p. 84) 分组。
集群状态	Amazon OpenSearch Service (OpenSearch Service) (p. 71)：群集的运行状况指示符。状态可以为绿色、黄色或红色。在分区级别，绿色表示所有分区都分配给集群中的节点；黄色表示分配了主分区，但未分配副本分区；红色表示至少一个索引的主分区和副本分区未分配。分片状态决定了索引状态，而索引状态决定了群集状态。
别名记录	规范名称记录。域名系统 (DNS) 中的一种资源记录 (p. 107)，用于指定该域名是另一个规范域名的别名。具体而言，它是 DNS 表中的一个条目，可用于将一个完全限定域名设置为另一个域名的别名。
Code Signing for Amazon IoT	一项服务，用于对您为 Amazon Web Services（Amazon）支持的任何 IoT 设备创建的代码进行签名。
投诉	一种事件，即不想收到电子邮件的收件人 (p. 106)在电子邮件客户端中单击 Mark as Spam (标记为垃圾邮件)，且 互联网服务提供商（ISP） (p. 95) 向 Amazon SES (p. 73) 发送通知。

复合查询	Amazon CloudSearch (p. 69)：使用 Amazon CloudSearch 结构化搜索语法指定多个搜索条件的搜索请求。
条件	IAM (p. 78)：有关权限的任何限制或详细信息。“D 适用的情况下，A 可以对 C 执行 B”语句中的条件是 D。 Amazon WAF (p. 82)：Amazon WAF 在针对 Amazon 资源 (p. 106)（如 Amazon CloudFront (p. 69) 分配）的 Web 请求中搜索的一组属性。条件可以包括 Web 请求源自的 IP 地址等值或请求标头中的值。根据指定的条件，您可以将 Amazon WAF 配置为允许或阻止Amazon资源的 Web 请求。
条件参数	See 映射.
配置 API	Amazon CloudSearch (p. 69)：您用于创建、配置和管理搜索域的 API 调用。
配置模板	一系列键值对，用于定义各种 Amazon 产品的参数，以便 Amazon Elastic Beanstalk (p. 77) 可以为环境预置这些产品。
一致性模型	服务用来实现高可用性的方法。例如，可能涉及在数据中心中多个服务器间复制数据。 See Also 最终一致性.
控制台	See Amazon Web Services Management Console.
整合账单	Amazon Organizations 服务中用于整合多个 Amazon Web Services 账户 的付款的功能。您创建一个包含您的 Amazon Web Services 账户 的组织，并使用组织的管理账户支付所有成员账户的款项。您可以查看组织中所有账户产生的 Amazon 成本的组合视图，并可以获得各个账户的详细成本报告。
容器	容器是包含应用程序代码和所有相关依赖关系的软件标准单位。
容器定义	容器定义指定了与在 Amazon ECS 上运行 容器 (p. 85) 相关的详细信息。具体来说，容器定义指定了要使用的容器映像以及为容器分配的 CPU 和内存等详细信息。容器定义作为 Amazon ECS 任务定义 (p. 113) 的一部分包含在内。
容器实例	容器实例是一个自行管理 EC2 实例 (p. 89) 或运行 Amazon Elastic Container Service (Amazon ECS) 容器代理并已注册到 集群 (p. 84) 的本地服务器或虚拟机。容器实例用作运行 Amazon ECS 工作负载的基础设施。
容器注册表	容器注册表是存储容器映像的存储库集合。一个示例是 Amazon Elastic Container Registry (Amazon ECR)。
内容分发网络 (CDN)	一项 Web 服务，通过由遍布全球的数据中心组成的网络，加速向您的用户分发静态和动态 Web 内容，例如，.html、.css、.js、媒体文件和图像文件。当用户请求获得内容时，该请求将路由到延迟 (时间滞延) 最短的数据中心。如果内容已经在延迟最短的位置，CDN 将立即提供它。否则，CDN 将从您指定的原始位置 (例如，Web 服务器或 Amazon S3 存储桶) 对其进行检索。可以通过某些 CDN 保护您的内容，方法是配置用户和数据中心、以及数据中心和原始位置之间的 HTTPS 连接。Amazon CloudFront 是 CDN 的一个示例。
上下文元数据	Amazon Personalize (p. 72)：您在事件（例如单击）发生时收集的有关用户浏览上下文（例如使用的设备或位置）的交互数据。上下文元数据可以改善新用户和现有用户的建议相关性。 See Also 交互数据集, event.
持续交付	一种软件开发实践，通过该实践可以自动构建、测试和准备代码更改以便投产。 See Also https://aws.amazon.com/devops/continuous-delivery/.
持续集成	一种软件开发实践，通过该实践，开发人员定期将代码更改合并到中央存储库，然后运行自动化构建和测试。

	See Also https://aws.amazon.com/devops/continuous-integration/.
冷却时间	一段时间，在此期间 Amazon EC2 Auto Scaling (p. 70) 不允许来自 Amazon CloudWatch (p. 69) 告警 (p. 68)的任何其他通知更改 自动扩缩组 (p. 75) 的所需大小。
核心节点	使用 Hadoop Distributed File System (HDFS) 运行 Hadoop 映射和缩减任务并存储数据的 EC2 实例 (p. 89)。Hadoop (p. 93)核心节点由 主节点 (p. 99) 管理，后者将 Hadoop 任务分配到节点并监控它们的状态。以核心节点的形式分配的 EC2 实例是为了整个任务流程运行必须分配的容量。由于核心节点负责存储数据，您无法从任务流程中移除它们。不过，您可以向正在运行的任务流程添加更多核心节点。 核心节点可以运行 DataNodes 和 TaskTracker 这两种 Hadoop 守护程序。
语料库	Amazon CloudSearch (p. 69)：要搜索的数据的集合。
覆盖	Amazon Personalize (p. 72)：一项评估指标，指明 Amazon Personalize 可能推荐使用您的模型的唯一项目占交互数据集和项目数据集总数的比例。要确保 Amazon Personalize 推荐您的更多项目，请使用覆盖率分数较高的模型。具有项目浏览功能的配方（例如用户-个性化）比不具备项目浏览功能的配方（例如热门程度-计数）具有更高的覆盖率。 See Also 指标, 物品数据集, 交互数据集, 物品浏览, 用户-个性化配方, 热门程度-计数配方.
凭证辅助程序	Amazon CodeCommit (p. 77)：一种程序，它存储用于存储库的凭证并在连接存储库时向 Git 提供这些凭证。Amazon CLI (p. 77) 提供在连接到 CodeCommit 存储库时可用于 Git 的凭证辅助程序。
凭证	也称作访问凭证 或安全凭证。在身份验证和授权中，系统使用凭证来识别谁在执行调用并决定是否允许请求的访问。在 Amazon 中，这些凭证通常是 访问密钥 ID (p. 67) 和 私有访问密钥 (p. 109)。
跨账户访问	允许一个 Amazon Web Services 账户 中的用户对另一个 Amazon Web Services 账户 (p. 68) 中的资源 (p. 106)进行有限、受控使用的过程。例如，在 Amazon CodeCommit (p. 77) 和 Amazon CodeDeploy (p. 77) 中，您可以配置跨账户访问，以便 Amazon Web Services 账户 A 中的用户可以访问由账户 B 创建的 CodeCommit 存储库。或者账户 A 在 Amazon CodePipeline (p. 77) 中创建的管道可以使用账户 B 创建的 CodeDeploy 资源。在 IAM (p. 78) 中，您可以使用role (p. 107)将对一个账户中 user (p. 114)的临时访问权限委托 (p. 88)给另一个账户中的资源。
跨区域复制	一种用于跨不同 Amazon Web Services 区域 (p. 106) 近乎实时复制数据的解决方案。
客户网关	Amazon VPC (p. 74) 管理的 VPN 隧道中您那一端的路由器或软件应用程序。客户网关的内部接口连接到您的家庭网络中的一个或多个设备。外部接口通过 VPN 隧道挂载到 虚拟私有网关（VGW） (p. 115)。
客户管理的策略	您在您的 Amazon Web Services 账户 (p. 68) 中创建并管理的 IAM (p. 78) 管理的策略 (p. 98)。
客户主密钥（CMK）	我们不再使用客户主密钥或 CMK。这些术语被替换为 Amazon KMS key（首次提及）和 KMS 密钥（后续提及）。有关更多信息，请参阅 KMS 密钥 (p. 97)。

D

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) |

O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

控制面板	See 服务运行状况控制面板.
数据一致性	一个概念，描述成功写入或更新数据的时间以及在所有 Amazon Web Services 区域 (p. 106) 中更新所有数据副本的时间。但是，要将数据传播到所有存储位置需要耗费一定的时间。为了支持各种应用程序要求，Amazon DynamoDB (p. 70) 同时支持最终一致性读取和强一致性读取。 See Also 最终一致性, 最终一致性读取, 强一致性读取.
数据节点	Amazon OpenSearch Service (OpenSearch Service) (p. 71)：一个保存数据并响应数据上传请求的 OpenSearch 实例。 See Also 专用主节点, node.
数据架构	See schema.
数据源	提供应用程序或数据库所需的信息的数据库、文件或存储库。例如，在 Amazon OpsWorks (p. 79) 中，有效的数据源包括堆栈的 MySQL 层或堆栈的 Amazon RDS (p. 73) 服务层的实例 (p. 95)。在 Amazon Redshift (p. 73) 中，有效的数据源将文本文件包含在 Amazon S3 (p. 74) 桶 (p. 83)、Amazon EMR (p. 71) 集群或远程主机（集群可通过 SSH 连接访问该主机）中。 See Also 数据源.
数据库引擎	在 数据库实例 (p. 87) 上运行的数据库软件和版本。
数据库名称	数据库实例 (p. 87) 中托管的数据库的名称。一个数据库实例可以托管多个数据库，但是，同一数据库实例托管的每一个数据库在该实例内必须有唯一名称。
dataset	Amazon Personalize (p. 72)：Amazon Personalize 所用数据的容器。Amazon Personalize 数据集有三种类型：用户、项目和交互。 See Also 交互数据集, 用户数据集, 物品数据集.
数据集组	Amazon Personalize (p. 72)：Amazon Personalize 组件的容器，包括数据集、事件跟踪器、解决方案、筛选器、活动和批量推理任务。数据集组将资源组织到独立的集合中，因此一个数据集组中的资源不会影响任何其他数据集组中的资源。 See Also dataset, 事件跟踪器, solution, 活动.
数据源	Amazon Machine Learning (p. 72)：包含有关输入数据的元数据的对象。Amazon ML 读取输入数据、计算其属性的描述性统计数据，并将统计数据与架构和其他信息一起存储为数据源对象的一部分。Amazon ML 使用数据源训练和评估机器学习模型并生成批量预测。 See Also 数据源.
数据库计算等级	用于运行实例的数据库计算平台的大小。
数据库实例	在云中运行的独立数据库环境。一个数据库实例可以包含多个由用户创建的数据库。
数据库实例标识符	用户为数据库实例提供的标识符。在 Amazon Web Services 区域 (p. 106) 中，该标识符对该用户必须是唯一的。
数据库参数组	应用于一个或多个 DB 实例 (p. 87)的数据库引擎参数值的容器。
数据库安全组	一种控制对 数据库实例 (p. 87) 的访问权限的方法。默认情况下，对数据库实例的网络访问权限是禁用的。对安全组 (p. 109)配置绑定流量后，这些规则将应用于与该组关联的所有数据库实例。
数据库快照	数据库实例 (p. 87)的用户启动时间点备份。
Dedicated Host	一个具有供用户专用的 EC2 实例 (p. 89)容量的物理服务器。

Dedicated Instance	一个在主机硬件级别物理隔离并在 VPC (p. 116) 内启动的实例 (p. 95)。
专用主节点	Amazon OpenSearch Service (OpenSearch Service) (p. 71)：一个执行集群管理任务但不保留数据或响应数据上传请求的 OpenSearch 实例。Amazon OpenSearch Service（OpenSearch Service）使用专用主节点来提高集群稳定性。 See Also 数据节点, node.
专用预留实例	您为确保在 VPC (p. 116) 中启动专用实例 (p. 88)时拥有足够容量而购买的选项。
委派	在单一 Amazon Web Services 账户 (p. 68) 中：授予 Amazon 用户 (p. 114)对 Amazon Web Services 账户 中资源 (p. 106)的访问权限。 在两个 Amazon Web Services 账户 之间：在拥有资源的账户（信任账户）与包含需要访问资源的用户的账户（受信任账户）之间设置信任。 See Also 信任策略.
删除标记	包含密钥和版本 ID 但不包含内容的对象。删除对象时，Amazon S3 (p. 74) 会自动将删除标记插入到版本控制的存储桶 (p. 83)中。
送达率	电子邮件能够到达其预期目的地的可能性。
传送	在一段时间内，经由 Amazon SES (p. 73) 发送，为 互联网服务提供商（ISP） (p. 95) 所接受以传送至收件人 (p. 106)的电子邮件数量。
拒绝	策略 (p. 103)语句的结果，以拒绝为结果，以便针对用户、组或角色明确禁止特定操作。显式拒绝优先于显式允许 (p. 68)。
部署配置	Amazon CodeDeploy (p. 77)：部署期间由服务使用的一组部署规则以及成功和失败条件。
部署组	Amazon CodeDeploy (p. 77)：一组单独标记的实例 (p. 95)或自动扩缩组 (p. 75)中的 EC2 实例 (p. 89)，或者两种实例。
详细监控	监控每 1 分钟产生一次的 Amazon 提供的指标。
说明属性	一个添加至参数、资源 (p. 106)、资源属性、映射和输出的属性，可帮助您记录 Amazon CloudFormation (p. 76) 模板元素。
维度	一个名称值对（例如，InstanceType=m1.small 或 EngineName=mysql），其中包含用于标识指标的其他信息。
开发论坛	Amazon 用户可在此发布技术问题和反馈以帮助加速其开发工作，也可在此与 Amazon 社区进行交流。有关更多信息，请参阅亚马逊云科技论坛。
分配	原始服务器（如 Amazon S3 (p. 74) 桶 (p. 83)）与 CloudFront (p. 69) 自动分配的域名之间的链接。借助该链接，CloudFront 能够识别您存储在 源服务器 (p. 101) 中的对象。
DKIM	域名密钥识别邮件。发件人为其电子邮件附加签名所用的标准。ISP 使用这些签名来验证电子邮件是否合法。有关更多信息，请参阅 https://tools.ietf.org/html/rfc6376。
DNS	See 域名系统.
Docker 镜像	作为 Docker 容器 (p. 85)的基础的分层文件系统模板。Docker 映像可包含特定的操作系统或应用程序。
文档	Amazon CloudSearch (p. 69)：可作为搜索结果返回的项目。每个文档都有一个字段集合，这些字段包含可供搜索或返回的数据。字段值可以是字符串，也可以是数字。每个文档都必须拥有唯一的 ID 和至少一个字段。

文档批处理 — Amazon CloudSearch (p. 69)：文档添加和删除操作的集合。您可以使用文档服务 API 提交批处理，以更新搜索域中的数据。

文档服务 API — Amazon CloudSearch (p. 69)：用于提交文档批处理以更新搜索域中数据的 API 调用。

文档服务终端节点 — Amazon CloudSearch (p. 69)：您向 Amazon CloudSearch 域发送文档更新时连接到的 URL。每个搜索域都拥有唯一的文档服务终端节点，文档服务终端节点在域的生命周期内保持不变。

domain — Amazon OpenSearch Service (OpenSearch Service) (p. 71)：由 Amazon OpenSearch Service（OpenSearch Service）端点公开的硬件、软件和数据。OpenSearch Service 域是一个围绕 OpenSearch 集群的服务包装程序。OpenSearch Service 域将封装用于处理 OpenSearch Service 请求的引擎实例、要搜索的索引数据、域的快照、访问策略和元数据。
See Also 集群, Elasticsearch.

域名系统 — 一项服务，可通过将人类可读的域名（例如 `www.example.com`）转换为数字 IP 地址（例如 192.0.2.1，计算机可利用这些地址互相连接）实现将互联网流量路由到网站。

捐赠按钮 — 一种用 HTML 编码的按钮，它为基于美国的 IRS 认证的 501(c)(3) 非营利性组织提供了一种简便、安全的募款方式。

DynamoDB 流 — 有关 Amazon DynamoDB (p. 70) 表中的项目更改的有序信息流。当您对表启用流时，DynamoDB 将捕获有关对表中的数据项目进行的每项修改的信息。
See Also Amazon DynamoDB Streams.

E

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) | O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

EBS — See Amazon Elastic Block Store (Amazon EBS).

EC2 — See Amazon EC2.

EC2 实例 — Amazon EC2 (p. 70) 服务中的计算实例 (p. 95)。其他 Amazon 服务使用术语 EC2 实例区分这些实例与其支持的其他类型的实例。

ECR — See Amazon Elastic Container Registry (Amazon ECR).

ECS — See Amazon Elastic Container Service (Amazon ECS).

边缘站点 — 一个数据中心，Amazon 服务用于执行特定于服务的操作。例如，CloudFront (p. 69) 使用边缘站点缓存内容的副本，因此内容更接近您的用户，并且无论其位置在何处，都可以更快地传递内容。Route 53 (p. 73) 使用边缘站点加快对公共 DNS 查询的响应速度。

EFS — See Amazon Elastic File System (Amazon EFS).

灵活应变 — 一家提供开源解决方案（包含 OpenSearch、Logstash、Kibana 和 Beats）的公司，这些方案可从任何源获取数据并实时搜索、分析和可视化数据。

Amazon OpenSearch Service（OpenSearch Service）是一项 Amazon 托管式服务，可用于在 Amazon Web Services 云 中部署、操作和扩展 OpenSearch。
See Also Amazon OpenSearch Service (OpenSearch Service), Elasticsearch.

Elastic Block Store	See Amazon Elastic Block Store (Amazon EBS).
弹性 IP 地址	您在 Amazon EC2 (p. 70)或 Amazon VPC (p. 74) 中分配并附加至实例 (p. 95)的固定（静态）IP 地址。弹性 IP 地址与您的账户（而非特定实例）相关联。之所以称其为弹性，是因为您可根据需求的变更方便地分配、挂载、取消挂载和释放它们。与传统静态 IP 地址不同，使用弹性 IP 地址，您可以快速将您的公有 IP 地址重新映射到其他实例，从而掩盖实例故障或可用区： (p. 76)故障。
弹性负载均衡	一项 Web 服务，可用于将传入流量分配到两个或多个 EC2 实例 (p. 89) 之间，从而提高应用程序的可用性。 See Also https://aws.amazon.com/elasticloadbalancing.
弹性网络接口	可挂载到 实例 (p. 95) 的附加网络接口。弹性网络接口包括一个主要私有 IP 地址、一个或多个次要私有 IP 地址、一个弹性 IP 地址（可选）、一个 MAC 地址、指定安全组 (p. 109)中的成员资格、一个描述以及一个源/目标检查标记。您可以创建一个弹性网络接口，将其附加到一个实例上，然后将其与实例分离再附加到另一个实例上。
Elasticsearch	一个开源、实时的分布式搜索和分析引擎，用于全文搜索、结构化搜索和分析。OpenSearch 由 Elastic 公司开发。 Amazon OpenSearch Service（OpenSearch Service）是一项 Amazon 托管式服务，可用于在 Amazon Web Services 云 中部署、操作和扩展 OpenSearch。 See Also Amazon OpenSearch Service (OpenSearch Service), 灵活应变.
EMR	See Amazon EMR.
加密	通过数学算法使未经授权的用户 (p. 114)难以理解数据。加密还为授权的用户提供了一种方法（例如密钥或密码），将更改后的数据转换回其原始状态。
加密上下文	一组键值对，其中包含与 Amazon Key Management Service (Amazon KMS) (p. 79) 加密信息关联的其他信息。
endpoint	指定作为某一 Web 服务入口点的主机和端口的 URL。每个 Web 服务请求都包含一个终端节点。大多数 Amazon 产品提供区域终端节点，以加快连接速度。 Amazon ElastiCache (p. 71)：缓存节点 (p. 83)的 DNS 名称。 Amazon RDS (p. 73)：数据库实例 (p. 87)的 DNS 名称。 Amazon CloudFormation (p. 76)：接收 HTTP 请求的服务器的 DNS 名称或 IP 地址。
终端端口	Amazon ElastiCache (p. 71)：缓存节点 (p. 83)使用的端口号。 Amazon RDS (p. 73)：数据库实例 (p. 87)使用的端口号。
信封加密	使用主密钥和数据密钥以便通过算法保护数据。主密钥用于加密和解密数据密钥，数据密钥用于加密和解密数据本身。
环境	Amazon Elastic Beanstalk (p. 77)：application (p. 75)的特定运行实例。该应用程序拥有别名记录，并包括应用程序版本和自定义配置（继承自默认容器类型）。 Amazon CodeDeploy (p. 77)：蓝/绿部署中的部署组内的实例。在蓝/绿部署开始时，部署组由原始环境中的实例组成。在部署结束时，部署组由替换环境中的实例组成。
环境配置	一组参数和配置，这些参数和配置用于定义环境及其相关资源的操作方式。
短暂存储	See 实例存储.

纪元	测量时间的起始日期。对于大多数 Unix 环境，纪元为 1970 年 1 月 1 日。
ETL	See 提取、转换和加载 (ETL).
评估	Amazon Machine Learning：衡量机器学习（ML）模型的预测性能的过程。 另一种存储 ML 模型评估的详细信息和结果的机器学习。
评估数据源	Amazon Machine Learning 用来评估机器学习模型的预测准确度的数据。
event	Amazon Personalize (p. 72)：您记录并上传到 Amazon Personalize 交互数据集的用户活动（如单击、购买或视频观看）。您可以实时单独记录事件，也可以批量记录和上传事件。 See Also dataset, 交互数据集.
事件跟踪器	Amazon Personalize (p. 72)：为实时记录的事件数据指定目标数据集组。当您实时记录事件时，需要提供事件跟踪器的 ID，以便 Amazon Personalize 了解在何处添加数据。 See Also 数据集组, event.
最终一致性	Amazon Web Services 用来实现高可用性的方法。这可能涉及在 Amazon 数据中心中多个服务器间复制数据。写入或更新数据时如返回 `Success`，则所有数据副本均得到更新。但是，要将数据传播到所有存储位置需要耗费一定的时间。数据最终将是一致的，但立即读取可能不会反映出这一变更。通常，在几秒钟内即可实现一致性。 See Also 数据一致性, 最终一致性读取, 强一致性读取.
最终一致性读取	仅从一个区域返回数据且可能不会显示最近的写入信息的读取过程。但是，如果您在短时间后重复读取请求，响应最终将返回最新的数据。 See Also 数据一致性, 最终一致性, 强一致性读取.
收回	CloudFront (p. 69) 在对象过期之前从边缘站点 (p. 89)删除对象的操作。如果边缘站点中的对象未获得频繁的请求，CloudFront 可能会将该对象移出（在对象过期日期之前删除对象）以便为更常用的对象腾出空间。
exbibyte (EiB)	百亿亿二进制字节的缩写形式。1 艾字节（EiB）为 2^60 字节，即 1.152921504606846976 百亿亿字节。1 exabyte (EB) 为 10^18 字节，即 1000000000000000000 字节。1024 EiB 为 1 zebibyte (ZiB) (p. 116)。
过期	对于 CloudFront (p. 69) 缓存，为 CloudFront 停止响应针对对象的用户请求的时间。如果您不使用标头或 CloudFront 分配 (p. 88)设置来指定对象在边缘站点 (p. 89)中的保留时间，则对象将在 24 小时后过期。当用户下次请求已过期的对象时，CloudFront 会将请求转发到源 (p. 101)。
显式展示	Amazon Personalize (p. 72)：您手动添加到 Amazon Personalize 交互数据集以影响未来推荐的项目列表。不同于隐式展示（Amazon Personalize 自动派生展示数据），您可以选择要在显式展示中包含的项目。 See Also 推荐, 交互数据集, 展示数据, 隐式展示.
显式启动的权限	授予特定 Amazon Web Services 账户 (p. 68) 的 Amazon Machine Image（AMI） (p. 72) 启动许可。
指数退避	一种增量增加重试尝试之间等待时间以便降低系统负载并增加重复请求成功可能性的策略。例如，客户端应用程序可能在尝试第一次重试前最多等待 400 毫秒，第二次重试最多等待 1600 毫秒，第三次重试最多等待 6400 毫秒（6.4 秒）。
expression	Amazon CloudSearch (p. 69)：用于控制搜索命中结果排序方式的数值表达式。您可以使用数值字段、其他排名表达式、文档的默认相关性分数、标准数值运算符和函数来构建 Amazon CloudSearch 表达式。使用 `sort` 选项在搜索请求中指定表达式时，将针对每个搜索命中结果评估该表达式，并根据其表达式值列出命中结果。

提取、转换和加载 (ETL)

用于集成来自多个源的数据的过程。从源收集数据 (提取)，将数据转换为适当的格式 (转换) 并将数据写入目标数据存储 (加载) 以进行分析和查询。

ETL 工具组合了这三种功能来整合数据并将数据从一个环境移动到另一个环境。Amazon Glue (p. 78) 是一种完全托管的 ETL 服务，用于发现和组织数据、转换数据并使其可用于搜索和分析。

F

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) | O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

分面

Amazon CloudSearch (p. 69)：一个索引字段，代表用于优化和筛选搜索结果的类别。

启用分面

Amazon CloudSearch (p. 69)：一个针对字段启用分面信息计算的索引字段选项。

FBL

See 反馈循环（FBL）.

功能转换

Amazon Machine Learning：从原始输入变量构造预测性更高的输入表示形式或“功能”以优化机器学习模型的学习和泛化能力的机器学习过程。也称作数据转换 或功能设计。

联合身份管理（FIM）

允许个人登录不同的网络或服务，使用相同的群组或个人凭证访问所有网络上的数据。借助 Amazon 中的身份联合验证，外部身份（联合用户）可以安全访问 Amazon Web Services 账户 (p. 68) 中的资源 (p. 106)，而不必创建 IAM 用户 (p. 114)。这些外部身份可能来自公司身份存储（如 LDAP 或 Windows Active Directory）或第三方（如 Login with Amazon、Facebook 或 Google）。Amazon 联合还支持 SAML 2.0。

联合身份用户

See 联合身份管理（FIM）.

联合身份验证

See 联合身份管理（FIM）.

反馈循环（FBL）

邮箱提供商（如 互联网服务提供商（ISP） (p. 95)）将 收件人 (p. 106) 的 投诉 (p. 84) 转发回 发件人 (p. 109) 的机制。

字段权重

文本字段在搜索索引中的相对重要程度。字段权重控制特定文本字段中的匹配项对文档的关联分数的影响程度。

筛选条件

在列出或描述 Amazon EC2 (p. 70) 资源 (p. 106)时，您指定的用于限制结果的条件。

筛选查询

一种筛选搜索结果而不对结果计分和排序产生影响的方式。通过 Amazon CloudSearch (p. 69) `fq` 参数指定。

FIM

See 联合身份管理（FIM）.

Firehose

See Amazon Kinesis Data Firehose.

格式版本

See 模板格式版本.

论坛

See 开发论坛.

函数

See 内部函数.

模糊搜索 一种使用近似字符串匹配（模糊匹配）来纠正键入错误和拼写错误的简单搜索查询。

G

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) | O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

地理空间搜索 一种使用以经度和纬度指定的位置来确定匹配项和对结果进行排序的搜索查询。

吉字节（GiB） 千兆二进制字节的缩写形式，1 GiB 为 2^30 字节，即 1073741824 字节。1 gigabyte (GB) 为 10^9 字节，即 1000000000 字节。1024 GiB 为 1 tebibyte (TiB) (p. 113)。

GitHub 使用 Git 进行版本控制的基于 Web 的存储库。

全局二级索引 一种带有可能与表中不同的分区键和排序键的索引。全局二级索引之所以称为全局，这是因为该索引上的查询可跨过所有分区，涵盖表中的所有数据。
See Also 本地二级索引.

授予 Amazon Key Management Service (Amazon KMS) (p. 79)：用于授予 Amazon 主体 (p. 104)长期权限以使用 KMS 密钥的机制。

授予令牌 一种标识符，允许授予 (p. 93)中的权限立即生效。

基本实际情况 机器学习 (ML) 模型训练过程中使用的观察，包括正确的目标属性值。要训练 ML 模型以预测房屋销售价格，输入观察通常将包含该地区以前的房屋销售价格。这些房屋的销售价格构成了基本实际情况。

group IAM (p. 78) 用户 (p. 114)的集合。可使用 IAM 组简化为多个用户指定和管理权限的过程。

H

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) | O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

Hadoop 通过使用集群和简单的编程模型启用对大数据的分布式处理的软件。有关详细信息，请参阅 http://hadoop.apache.org。

硬退回邮件 持久性的电子邮件传送失败，例如“邮箱不存在。”

硬件 VPN Internet 上基于硬件的 IPsec VPN 连接。

运行状况检查 检查 Amazon EC2 Auto Scaling (p. 70) 群组中每个实例健康状况的系统调用。

高质量电子邮件 收件人认为有价值和想要收取的电子邮件。对不同的收件人而言，有价值的事物各不相同，它们可能是报价、订单确认函、收据、新闻通讯等等。

突出显示 Amazon CloudSearch (p. 69)：随搜索结果返回的摘要，它们显示了搜索词出现在匹配文档中的哪些文本位置。

启用突出显示 Amazon CloudSearch (p. 69)：一个使字段中的匹配项突出显示的索引字段选项。

命中结果	与搜索请求中指定的标准相匹配的文档。也称作搜索结果。
HMAC	基于哈希的消息身份验证代码。用于计算消息身份验证代码 (MAC) 的特定构建涉及加密哈希函数与密钥的组合。您可以将它用于同时确认数据的完整性和信息的真实性。Amazon 使用标准的加密哈希算法（如 SHA-256）计算 HMAC。
托管区域	Amazon Route 53 (p. 73) 托管的资源记录 (p. 107)集的集合。与传统的 DNS 区域文件类似，托管区域表示在单一域名下统一管理的记录集合。
HRNN	Amazon Personalize (p. 72)：一种分层循环神经网络机器学习算法，用于对用户行为的变化建模，并预测用户在个人推荐应用程序中可能与之交互的项目。
HTTP 查询	See 查询.
HVM 虚拟化	硬件虚拟机虚拟化。允许访客虚拟机如同在本地硬件平台上一样运行，唯一不同的是仍然使用半虚拟化（PV）网络和存储驱动程序提高性能。 See Also 半虚拟化.

I

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) | O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

IAM	See Amazon Identity and Access Management (IAM).
IAM Identity Center	See Amazon IAM Identity Center (successor to Amazon Single Sign-On).
IAM 组	See group.
IAM policy simulator	See 策略模拟器.
IAM 角色	See role.
IAM 用户	See user.
Identity and Access Management	See Amazon Identity and Access Management (IAM).
身份提供商（IdP）	一个包含有关外部身份提供商的元数据的 IAM (p. 78) 实体。
IdP	See 身份提供商（IdP）.
image	See Amazon Machine Image（AMI）.
导入/导出站	在 Amazon S3 (p. 74) 中上传或下载数据的机器。
导入日志	一种报告，包含有关 Amazon Import/Export (p. 78) 如何处理您的数据的详细信息。
隐式展示	Amazon Personalize (p. 72)：您的应用程序向用户显示的推荐。不同于显式展示（手动记录每个展示），Amazon Personalize 自动从您的推荐数据派生隐式展示。 See Also 推荐, 展示数据, 显式展示.
展示数据	Amazon Personalize (p. 72)：当用户与特定物品进行交互（例如单击、观看或购买）时，您向其展示的物品列表。Amazon Personalize 根据用户选择或忽略同一物品的频率，使用展示数据计算新物品对用户的相关性。

	See Also 显式展示, 隐式展示.
就地部署	CodeDeploy：一种部署方法，即停止部署组中每个实例上的应用程序，安装最新的应用程序修订版，然后启动并验证应用程序的新版本。您可以选择使用负载均衡器，以便在其部署期间取消注册每个实例，然后在部署完成后让其恢复服务。
index	See 搜索索引.
索引字段	包含在 Amazon CloudSearch (p. 69) 域索引中的名称值对。索引字段可包含文本或数值数据、日期或位置。
索引选项	定义 Amazon CloudSearch (p. 69) 域索引字段、文档数据如何映射到这些索引字段以及如何使用索引字段的配置设置。
内联策略	嵌入在单个 IAM user (p. 114)、group (p. 93)或role (p. 107)中的 IAM (p. 78) 策略 (p. 103)。
输入数据	Amazon Machine Learning：您向 Amazon Machine Learning 提供的用来训练和评估机器学习模型并生成预测的观察。
实例	在 Amazon Web Services 云 中作为虚拟服务器运行的 Amazon Machine Image（AMI） (p. 72) 的副本。
实例系列	使用存储或 CPU 能力的通用 实例类型 (p. 95) 分组。
实例组	每个 Hadoop (p. 93) 集群包含一个主实例组（带有一个 主节点 (p. 99)）、一个核心实例组（带有一个或多个 核心节点 (p. 86)）和一个可选的 任务节点 (p. 113) 实例组（可以包含任意数量的任务节点）。
实例配置文件	一个在启动时将 IAM (p. 78) role (p. 107)信息传递给 EC2 实例 (p. 89) 的容器。
实例存储	以物理方式挂载到 EC2 实例 (p. 89) 的主机的磁盘存储，因而具有与该实例相同的生命周期。当实例终止时，实例存储中的所有数据都将丢失。
实例存储支持的 AMI	一种 Amazon Machine Image（AMI） (p. 72)，其实例 (p. 95)使用 实例存储 (p. 95) volume (p. 116) 作为根设备。将此实例与由 Amazon EBS (p. 70) 支持的 AMI 启动的实例进行比较，后者使用 Amazon EBS 卷作为根设备。
实例类型	定义内存、CPU、存储容量以及实例 (p. 95)使用成本的规范。某些实例类型适用于标准应用程序，而另一些则适用于 CPU 密集型、内存密集型应用程序。
交互数据集	Amazon Personalize (p. 72)：用于保存从用户和物品（称为事件）之间的交互中收集的历史数据和实时数据的容器。交互数据可以包括展示数据和上下文元数据。See Also dataset, event, 展示数据, 上下文元数据.
互联网网关	将网络连接到 Internet。您可以将 VPC (p. 116) 以外的 IP 地址的流量路由到 Internet 网关。
互联网服务提供商（ISP）	为订阅者提供 Internet 访问的公司。许多 ISP 也是邮箱提供商 (p. 98)。邮箱提供商有时也称作 ISP，即便他们仅提供邮箱服务。
内部函数	Amazon CloudFormation (p. 76) 模板中的一种特殊操作，可将值分配给直至运行时才可用的属性。这些函数遵循 Fn::Attribute 格式，例如 `Fn::GetAtt`。内部函数的参数可以是实际参数、虚拟参数或其他内部函数的输出。
IP 地址	一个数字地址（例如 192.0.2.44），供联网设备用来通过 Internet 协议 (IP) 相互通信。每个 EC2 实例 (p. 89) 在启动时都被分配了两个 IP 地址，即私有 IP 地址（遵循 RFC 1918）和公有 IP 地址，它们可通过网络地址转换（NAT (p. 100)）直接相互映射。VPC (p. 74) 中启动的实例只分配了私有 IP 地址。在默认 VPC 中启动的实例会分配到一个私有 IP 地址和一个公有 IP 地址。

IP 匹配条件
: Amazon WAF (p. 82)：一个指定 Web 请求源自的 IP 地址或 IP 地址范围的属性。根据指定的 IP 地址，可以将 Amazon WAF 配置为允许或阻止针对 Amazon 资源 (p. 106)（如 Amazon CloudFront (p. 69) 分配）的 Web 请求。

ISP
: See 互联网服务提供商（ISP）.

发布者
: 编写策略 (p. 103)以授予资源 (p. 106)权限的人员。发布者（按定义）通常指资源拥有者。Amazon 不允许 Amazon SQS (p. 73) 用户为他们不拥有的资源创建策略。如果 John 是资源拥有者，那么当他提交他编写的策略为该资源授予权限时，Amazon 会对 John 的身份进行认证。

item
: 一组属性，这些属性可在所有其他项目之间进行唯一标识。Amazon DynamoDB (p. 70) 中的项目在很多方面都类似于其他数据库系统中的行、记录或元组。

物品浏览
: Amazon Personalize (p. 72)：Amazon Personalize 用于测试不同项目推荐的过程，包括在没有或只有很少交互数据的情况下推荐新项目，并了解用户的反响。您可以在活动级别为使用用户-个性化配方创建的解决方案版本配置物品浏览。
See Also 推荐, 活动, 解决方案版本, 用户-个性化配方.

物品间相似度 (SIMS) 配方
: Amazon Personalize (p. 72)：一个 RELATED_ITEMS 配方，使用交互数据集中的数据为类似于指定物品的物品生成建议。SIMS 配方根据用户与物品进行交互的方式而不是匹配物品元数据（如价格或期限）计算相似度。
See Also recipe, RELATED_ITEMS 配方, 交互数据集.

物品数据集
: Amazon Personalize (p. 72)：用于保存物品相关元数据（例如价格、种类或可用性）的容器。
See Also dataset.

J

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) | O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

任务流程
: Amazon EMR (p. 71)：指定要对数据执行所有功能的一个或多个步骤 (p. 111)。

任务 ID
: 由五个字符组成的字母数字字符串，用于唯一地标识发运的 Amazon Import/Export (p. 78) 存储设备。Amazon 发出任务 ID 来响应 `CREATE JOB` 电子邮件命令。

任务前缀
: 一个可选字符串，可将它添加到 Amazon Import/Export (p. 78) 日志文件名的开头来防止与同名对象发生冲突。
See Also 键前缀.

JSON
: JavaScript 对象表示法。一种轻量级数据交换格式。想要了解更多有关 JSON 的信息，请参阅 http://www.json.org/。

垃圾邮件文件夹
: 用于收集各种筛选条件认定无甚价值的电子邮件的位置，这些电子邮件不会进入收件人 (p. 106)的收件箱，但仍可为收件人访问。也称作 垃圾电子邮件 (p. 111) 或批量文件夹。

K

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) |

O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

key

标识 Amazon Web Services 账户 (p. 68) 或 Amazon user (p. 114) 的凭证（例如 Amazon 私有访问密钥 (p. 109)）。

Amazon Simple Storage Service (Amazon S3) (p. 74)，Amazon EMR (p. 71)：对象在桶 (p. 83)内的唯一标识符。存储桶内的每个对象都只能有一个键。由于存储桶和键一起唯一地标识每个对象，因此可将 Amazon S3 视为一种存储桶 + 键与对象本身间的基本数据映射。将 Web 服务终端节点、存储桶名称和键组合在一起可唯一地寻址 Amazon S3 中的每个对象，例如在 `http://doc.s3.amazonaws.com/2006-03-01/AmazonS3.wsdl` 中，`doc` 是存储桶的名称，`2006-03-01/AmazonS3.wsdl` 是键。

Amazon Import/Export (p. 78)：Amazon S3 中对象的名称。它是一个 Unicode 字符序列，其 UTF-8 编码不能超过 1024 个字节。如果键（如 logPrefix + import-log-JOBID）超过 1024 个字节，Amazon Elastic Beanstalk (p. 77) 将返回 `InvalidManifestField` 错误。

IAM (p. 78)：在 策略 (p. 103) 中，作为限制访问基础的特定特征（如当前时间或请求者的 IP 地址）。

标记资源：一种常见的标签 (p. 113)标签，行为类似于更具体的标签值的类别。例如，您可能有一个标签键为 Owner 和标签值为 Jan 的 EC2 实例 (p. 89)。您最多可以使用 10 个键值对标记 Amazon 资源 (p. 106)。并非所有 Amazon 资源都可添加标签。

密钥对

一组用于以电子方式证明个人身份的安全凭证。密钥对包含私有密钥和公有密钥。

键前缀

字符串是对象键名称的子集，从第一个字符开始。前缀可以是任意长度，最长为对象键名称的最大长度（1024 字节）。

kibibyte（KiB）

千位二进制字节的缩写形式，1 KiB 为 2^10 字节，即 1024 字节。1 kilobyte (KB) 为 10^3 字节，即 1000 字节。1024 KiB 为 1 mebibyte (MiB) (p. 99)。

KMS

See Amazon Key Management Service (Amazon KMS).

KMS 密钥

Amazon Key Management Service 中的主要资源。通常，KMS 密钥完全在 KMS 中创建、使用和删除。KMS 支持将对称和非对称 KMS 密钥用于加密和签名。KMS 密钥可以是客户管理的、Amazon 托管的或 Amazon 所有的密钥。有关更多信息，请参阅 Amazon Key Management Service 开发人员指南中的 Amazon KMS keys。

L

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) | O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

标记的数据

在机器学习中，已知道其目标或“正确”答案的数据。

启动配置

一组说明性参数，用于在 Amazon EC2 Auto Scaling (p. 70) 活动中创建新的 EC2 实例 (p. 89)。

自动扩缩组 (p. 75) 用于启动新的 EC2 实例的模板。启动配置包含 Amazon Machine Image（AMI） (p. 72) ID、实例类型、密钥对、安全组 (p. 109)、块设备映射以及其他配置设置等信息。

启动权限

一个允许用户启动 AMI 的 Amazon Machine Image（AMI） (p. 72) 属性。

生命周期 — 包含在 自动扩缩组 (p. 75)中的 EC2 实例 (p. 89)的生命周期状态。EC2 实例在其生命周期中将经过许多状态；其中包括正在等待、正在运行、正在终止和已终止。

生命周期操作 — 一个可由 Auto Scaling 暂停的操作，例如启动或终止 EC2 实例。

生命周期挂钩 — 用于在 Auto Scaling 启动或终止 EC2 实例后将其暂停的功能，以便在实例未处于可用状态时执行自定义操作。

负载均衡器 — 与一组端口结合使用的 DNS 名称，它们一起为针对您的应用程序的所有请求提供了一个目标。负载均衡器可将流量分配到 区域 (p. 106) 中每个 可用区： (p. 76) 上的多个应用程序实例。负载均衡器可跨越在其中启动 Amazon EC2 (p. 70) 实例的一个 Amazon Web Services 区域 内的多个可用区。但是，负载均衡器不能跨多个区域。

本地二级索引 — 一种分区键与表中的相同但排序键与表中的不同的索引。本地二级索引之所以称为“本地”，是因为该索引的每个分区的范围都限定为具有相同分区键值的表分区。See Also 本地二级索引.

逻辑名称 — 标识资源 (p. 106)、映射 (p. 98)、参数或输出的 Amazon CloudFormation (p. 76) 模板中的一个区分大小写的唯一字符串。在 Amazon CloudFormation 模板中，每个参数、资源 (p. 106)、属性、映射和输出都必须用一个唯一的逻辑名称声明。当您使用 `Ref` 函数取消引用这些项时，请使用逻辑名称。

M

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) | O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

邮件传输代理 (MTA) — 借助客户端-服务器架构将电子邮件从一台计算机传输至另一台计算机的软件。

邮箱提供商 — 提供电子邮件邮箱托管服务的组织。邮箱提供商有时也称作互联网服务提供商（ISP）(p. 95)，即便他们仅提供邮箱服务。

邮箱模拟器 — 一组电子邮件地址，可用于测试基于 Amazon SES (p. 73) 的电子邮件发送应用程序，而不向实际收件人发送电子邮件。每个电子邮件地址代表某一特定场景（例如退回邮件或投诉），并生成特定于该场景的典型响应。

主路由表 — 任何新的 VPC (p. 116) subnet (p. 112)用于路由的默认路由表 (p. 107)。您可以将子网与您选择的不同路由表关联起来。也可以更改作为主路由表的路由表。

管理的策略 — 一个独立的 IAM (p. 78) 策略 (p. 103)，可附加到 IAM 账户 (p. 68) 中的多个用户 (p. 114)、组 (p. 93)和角色 (p. 107)。托管式策略可以是 Amazon 管理的策略（由 Amazon 创建和管理）或客户管理的策略（您在 Amazon Web Services 账户 中创建和管理）。

清单 — 当发送创建任务 请求以执行导入或导出操作时，您需要在称作清单的文本文件中描述您的任务。清单文件是一种 YAML 格式文件，用于指定如何在存储设备和 Amazon Web Services 云 之间传输数据。

清单文件 — Amazon Machine Learning：用于描述批量预测的文件。清单文件将每个输入数据文件与其关联的批量预测结果相关联。它存储在 Amazon S3 输出位置。

映射 — 一种将条件参数值添加到 Amazon CloudFormation (p. 76) 模板中的方法。您在模板中可选的 Mappings 部分指定映射，并使用 `FN::FindInMap` 函数检索所需的值。

标记 — See 分页标记.

主节点
在 Amazon Machine Image（AMI）(p. 72) 上运行的进程，负责跟踪其核心节点和任务节点完成的工作。

最高价
为启动一个或多个竞价型实例 (p. 111)所需支付的最高价格。如果您的最高价高于当前 Spot 价格 (p. 111)，并且满足您的限制，则 Amazon EC2 (p. 70) 会代表您启动实例。

最大发送速率
使用 Amazon SES (p. 73) 每秒可发送的电子邮件的最大数量。

前 25 项中平均倒数排名
Amazon Personalize (p. 72)：一项评估指标，用于评估模型中排名最高推荐的相关性。Amazon Personalize 在针对所有推荐请求的前 25 个推荐中最相关的推荐进行排名时，使用模型的平均准确度计算此指标。
See Also 指标, 推荐.

mebibyte (MiB)
百万二进制字节的缩写形式。1 兆字节（MiB）为 2^20 或 1.048576 百万字节。1 megabyte (MB) 为 10^6 字节，即 1000000 字节。1024 MiB 为 1 吉字节（GiB）(p. 93)。

成员资源
See 资源.

消息 ID
Amazon Simple Email Service (Amazon SES) (p. 73)：分配给所发送的每封电子邮件的唯一标识符。

Amazon Simple Queue Service (Amazon SQS) (p. 73)：您向队列发送消息时返回的标识符。

元数据
有关其他数据或对象的信息。在 Amazon Simple Storage Service (Amazon S3) (p. 74) 和 Amazon EMR (p. 71) 中，元数据采用描述对象的名称值对的形式。其中包括默认元数据（如上次修改日期）和标准 HTTP 元数据（如 Content-Type）。用户还可以在存储对象时指定自定义元数据。在 Amazon EC2 (p. 70) 中，元数据包括有关 EC2 实例 (p. 89)的数据，该实例可检索这些数据以确认自身状况，如实例类型或 IP 地址。

指标
一个时间序列数据元素，由一个命名空间 (p. 100)、一个指标名称和零到十个维度的唯一组合定义。从其得出的指标和统计数据是 Amazon CloudWatch (p. 69) 的基础。

指标
Amazon Personalize (p. 72)：Amazon Personalize 在您训练模型时生成的评估数据。您可以使用指标评估模型的性能、查看修改解决方案配置的效果，以及比较使用相同训练数据但使用不同配方创建的解决方案之间的结果。
See Also solution, recipe.

指标名称
主要的指标标识符，与 命名空间 (p. 100) 和可选维度搭配使用。

MFA
See 多重身份验证（MFA）.

微型实例
一种 EC2 实例 (p. 89)，如果您只偶尔会有高 CPU 的活动，则使用该类型会更合算。

MIME
See 多用途 Internet 邮件扩展 (MIME).

ML 模型
在机器学习（ML）中，是指通过在数据中查找模式来生成预测的数学模型。Amazon Machine Learning 支持三种类型的机器学习（ML）模型：二进制分类、多类别分类和回归。也称作预测模型。
See Also 二进制分类模型, 多类别分类模型, 回归模型.

MTA
See 邮件传输代理 (MTA).

多可用区部署
一个主 数据库实例 (p. 87)，在不同的 可用区： (p. 76) 拥有同步备用副本。主数据库实例可以跨可用区同步复制到备用副本。

多类别分类模型	一种预测属于有限的、预定义的允许值集的值的机器学习模型。例如，“该产品是书、影片还是服装？”
多重身份验证（MFA）	一种可选的 Amazon Web Services 账户 (p. 68) 安全功能。启用 Amazon MFA 后，无论何时访问安全的 Amazon 网页或 Amazon Web Services Management Console (p. 79)，除了您的登录凭证之外，您还必须提供一个六位数的一次性代码。您从物理拥有的身份验证设备中获取此一次性代码。 See Also https://aws.amazon.com/mfa/.
多值属性	具有多个值的属性。
分段上传	一种功能，可用于将单个对象作为一组分段上传。
多用途 Internet 邮件扩展 (MIME)	一种互联网标准，可扩展电子邮件协议以包含非 ASCII 文本和非文本元素（如附件）。
Multitool	一种为管理大型数据集提供简单命令行界面的级联应用程序。

否

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) | O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

命名空间	一种提供其所存放项目（名称、技术术语或单词）的上下文并消除位于不同命名空间中的同名项目之歧义的抽象容器。
NAT	网络地址转换。一种策略，其中当数据包跨流量路由设备传输时将一个或多个 IP 地址映射到另一个 IP 地址。此策略通常用于限制与私有实例的 Internet 通信并允许传出流量。 See Also 网络地址转换和协议转换, NAT 网关, NAT 实例.
NAT 网关	一种由 Amazon 托管的 NAT (p. 100) 设备，可在私有subnet (p. 112)中执行网络地址转换以保护入站互联网流量。NAT 网关同时使用 NAT 和端口地址转换。 See Also NAT 实例.
NAT 实例	一种由用户配置的 NAT (p. 100) 设备，可在 VPC (p. 116) 公共subnet (p. 112)中执行网络地址转换以保护入站 Internet 流量。 See Also NAT 网关.
网络 ACL	一个可选安全层，可作为防火墙来控制进出subnet (p. 112)的流量。一个网络 ACL (p. 67) 可关联多个子网，但在某一时刻，一个子网仅可关联一个网络 ACL。
网络地址转换和协议转换	(NAT (p. 100)-PT) RFC 2766 中定义的一种 Internet 协议标准。 See Also NAT 实例, NAT 网关.
n 元处理器	一种处理 n 元转换的处理器。 See Also n 元转换.
n 元转换	Amazon Machine Learning：一种帮助执行文本字符串分析的转换。n 元转换通过以下方式将文本变量用作输入和输出字符串：在文本上方滑动大小为 n 个词的窗口（其中 n 由用户指定）并输出大小为 n 和更小的每个单词字符串。例如，指定窗口大小为 2 的 n 元转换将返回所有 2 个单词组合以及所有单个单词。
NICE Desktop Cloud Visualization	一种远程可视化技术，可让用户安全地连接到在远程高性能服务器上托管的图形密集型 3D 应用程序。
node	Amazon OpenSearch Service (OpenSearch Service) (p. 71)：一个 OpenSearch 实例。节点可以是数据实例或专用主实例。

See Also 专用主节点.

NoEcho
Amazon CloudFormation (p. 76) 参数的一个属性，可阻止报告模板参数的名称和值（默认行为是报告）。声明 `NoEcho` 属性将导致 `cfn-describe-stacks` 命令所产生的报告中用星号代替参数值。

K (5/10/25) 项标准化折扣累计增益（NCDG）
Amazon Personalize (p. 72)：一项评估指标，指明了模型中排名较高的推荐的相关性，其中 K 是样本大小，即 5、10 或 25 个推荐。Amazon Personalize 按照以下方式计算此值：根据推荐在排名列表中的位置为其分配权重，其中每个推荐都根据其位置给予一定的折扣（给予较低的权重）。K 项标准化折扣累计增益假定列表中排名较低的推荐与列表中排名较高的推荐相比相关性更低。
See Also 指标, 推荐.

NoSQL
高度可用的、可扩展的并且已针对高性能进行优化的非关系数据库系统。有别于关系模型，NoSQL 数据库（如 Amazon DynamoDB (p. 70)）使用替代模型进行数据管理，例如键-值对或文档存储。

空对象
空对象是其版本 ID 为空的对象。当存储桶的版本控制 (p. 115)暂停时，Amazon S3 (p. 74) 会向该 桶 (p. 83) 中添加一个空对象。对于存储桶中的每个键，可能只有一个空对象。

传递次数
允许 Amazon Machine Learning 使用相同的数据记录来训练机器学习模型的次数。

O

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) | O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

object
Amazon Simple Storage Service (Amazon S3) (p. 74)：Amazon S3 中存储的基本实体类型。对象由对象数据和元数据组成。数据部分对 Amazon S3 是不透明的。

Amazon CloudFront (p. 69)：可通过 HTTP 或 RTMP 版本提供的实体。

观测值
Amazon Machine Learning：Amazon Machine Learning (Amazon ML) 用来训练机器学习模型如何预测或生成预测的单个数据实例。Amazon ML 输入数据文件中的每个行均为一个观察。

个按需实例
按小时或分钟（至少 60 分钟）收取计算容量费用的 Amazon EC2 (p. 70) 定价选项，无需做出长期承诺。

operation
一个 API 函数。也称作动作。

乐观锁
一种策略，用于确保要更新的项目在更新前未由其他人修改。对于 Amazon DynamoDB (p. 70)，Amazon SDK 提供乐观锁支持。

组织
Amazon Organizations (p. 80)：您创建用于整合和管理您的 Amazon Web Services 账户 的实体。一个组织有一个管理账户以及零个或多个成员账户。

组织单位
Amazon Organizations (p. 80)：组织的 根 (p. 107) 中的账户的容器。一个组织单位 (OU) 可包含其他 OU。

源访问身份
也称作 OAI。当使用 Amazon CloudFront (p. 69) 提供以 Amazon S3 (p. 74) 桶 (p. 83) 作为源的内容时，指一种虚拟身份，用于要求用户通过 CloudFront URL 而非 Amazon S3 URL 访问您的内容。通常用于 CloudFront 私有内容 (p. 104)。

源服务器
Amazon S3 (p. 74) 桶 (p. 83) 或自定义源，包含您通过 CloudFront (p. 69) 传输的内容的明确原始版本。

原始环境	CodeDeploy 蓝/绿部署开始时部署组中的实例。
OSB 转换	正交稀疏二元转换。机器学习中的一种转换，可帮助进行文本字符串分析，并且是 n 元转换的替代。OSB 转换通过以下方式生成：滑动文本上方的大小为 n 个词的窗口并输出包含窗口中的第一个词的每个单词对。 See Also n 元转换.
OU	See 组织单位.
输出位置	Amazon Machine Learning：用于存储批量预测结果的 Amazon S3 位置。

P

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) | O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

分页	通过以多个单独的小部分返回一大批记录来响应 API 请求的过程。分页可在以下情况下发生： • 客户端将最大返回记录数设置为一个小于总记录数的值。 • 服务具有一个小于总记录数的默认最大返回记录数。 在对 API 响应进行分页时，此服务将发送大批记录中的一小部分记录和一个指示多个记录可用的分页标记。客户端将此分页标记包含在后续 API 请求中，并且服务将发送下一记录子集作为响应。此操作将继续，直到服务发送某个记录子集作为响应且没有分页标记，这指示已发送所有记录。
分页标记	一个指示 API 响应包含大批记录中的某个记录子集的标记。客户端可在后续 API 请求中返回此标记以检索下一个记录子集，直到服务发送某个记录子集作为响应且没有分页标记，这指示已发送所有记录。 See Also 分页.
付费 AMI	在 Amazon Marketplace (p. 79) 上出售给其他 Amazon EC2 (p. 70) 用户的 Amazon Machine Image（AMI） (p. 72)。
半虚拟化	See 半虚拟化.
part	分段上传请求中的对象数据的连续部分。
分区键	一个简单主键，由一个属性（也称作哈希属性）构成。 See Also 分区键, 排序键.
PAT	端口地址转换。
pebibyte (PiB)	千万亿二进制字节的缩写形式，1 PiB 为 2^50 字节，即 1125899906842624 字节。1 petabyte (PB) 为 10^15 字节，即 1000000000000000 字节。1024 PiB 为 1 exbibyte (EiB) (p. 91)。
周期	See 采样周期.
许可	策略 (p. 103)中允许或拒绝访问特定资源 (p. 106)的语句。您可以通过以下方式声明任意权限：“A has permission to do B to C.”例如，Jane (A) 有权从 John 的 Amazon SQS (p. 73) 队列 (C) 读取消息 (B)。无论 Jane 何时向 Amazon SQS 发送使用 John 的队列的请求，该服务均会检查她是否具有权限。它还进一步检查请求是否满足 John 在权限中设定的条件。

持久性存储	一种数据存储解决方案，其中，数据在被删除前将保持不变。Amazon (p. 74) 中的选项包括：Amazon S3 (p. 74)、Amazon RDS (p. 73)、Amazon DynamoDB (p. 70) 和其他服务。
PERSONALIZED_RANKING recipes	Amazon Personalize (p. 72)：根据用户的预测兴趣按排名顺序提供物品推荐的配方。 See Also recipe, 推荐, 个性化-排名配方, 热门程度-计数配方.
个性化-排名配方	Amazon Personalize (p. 72)：一种 PERSONALIZED_RANKING 配方，根据特定用户的预测兴趣级别对您提供的一系列物品进行排名。使用个性化-排名配方可创建针对特定用户个性化的物品或有序搜索结果的精选列表。 See Also recipe, PERSONALIZED_RANKING recipes.
实体名称	创建堆栈 (p. 111)时分配给每个资源 (p. 106)的 Amazon CloudFormation (p. 76) 的唯一标签。某些 Amazon CloudFormation 命令接受实体名称用作 `--physical-name` 参数的值。
管道	Amazon CodePipeline (p. 77)：通过发布程序定义软件更改方式的工作流程结构。
明文	尚未加密 (p. 90)的信息（与密文 (p. 84)相对）。
策略	IAM (p. 78)：定义应用于用户、组或角色的权限的文档；权限进而确定用户可以在 Amazon 中执行的操作。策略通常允许 (p. 68)访问特定操作，可以选择授权允许对特定资源 (p. 106)（例如，EC2 实例 (p. 89)或 Amazon S3 (p. 74) 存储桶 (p. 83)）执行操作。策略还可以显式拒绝 (p. 88)访问。 Amazon EC2 Auto Scaling (p. 70)：存储启动或终止自动扩缩组实例所需信息的对象。运行该策略会导致实例启动或终止。您可以配置告警 (p. 68)以调用 Auto Scaling 策略。
策略生成器	IAM (p. 78) Amazon Web Services Management Console (p. 79)中的一种工具，可帮助您通过选择可用选项列表中的元素来生成策略 (p. 103)。
策略模拟器	IAM (p. 78) Amazon Web Services Management Console (p. 79)中的一种工具，可帮助您对策略 (p. 103)进行测试和故障排除，以便您能查看策略在实际场景中所起的作用。
策略验证器	IAM (p. 78) Amazon Web Services Management Console (p. 79)中的一种工具，可检查现有 IAM 访问控制策略 (p. 103)以确保它们符合 IAM policy 语法。
热门程度-计数配方	Amazon Personalize (p. 72)：一种 USER_Producation 配方，用于推荐与独立用户交互最多的物品。 See Also recipe, USER_PERSONALIZATION 配方.
K (5/10/25) 项精度	Amazon Personalize (p. 72)：一项评估指标，指明了根据样本大小 K（5、10 或 25）个推荐，您模型的推荐的相关性。Amazon Personalize 将按照以下方法计算此指标：前 K 个推荐中的相关推荐数量除以 K，其中 K 为 5、10 或 25。 See Also 指标, 推荐.
prefix	See 任务前缀.
Premium Support	A one-on-one, fast-response support channel that Amazon customers can subscribe to for support for Amazon infrastructure services. See Also https://aws.amazon.com/premiumsupport/.
预签名 URL	一个使用查询字符串身份验证 (p. 105)的 Web 地址。
主键	一个或两个属性，可唯一标识 Amazon DynamoDB (p. 70) 表中的每个项目，以确保任意两个项目不具有相同的键。

	See Also 分区键, 排序键.
主分区	See 分片.
委托人	获得策略 (p. 103)中定义的权限的user (p. 114)、服务或账户 (p. 68)。委托人是“A 有权对 C 执行 B”语句中的 A。
私有内容	当使用 Amazon CloudFront (p. 69) 提供以 Amazon S3 (p. 74) 桶 (p. 83)作为源的内容时时，指一种为通过要求用户使用签名的 URL 来控制对内容的访问的方法。签名的 URL 可根据当前日期和时间和/或请求源 IP 地址来限制用户访问。
私有 IP 地址	一个私有数字地址（例如 192.0.2.44），供联网设备用来通过 Internet 协议 (IP) 相互通信。每个 EC2 实例 (p. 89) 在启动时都被分配了两个 IP 地址，即私有地址（遵循 RFC 1918）和公有地址，它们可通过网络地址转换（NAT (p. 100)）直接相互映射。例外：在 Amazon VPC (p. 74) 中启动的实例只分配有私有 IP 地址。
私有子网	一种 VPC (p. 116) subnet (p. 112)，其实例不能从互联网访问。
产品代码	Amazon在您将产品提交到 Amazon Marketplace (p. 79) 时提供的标识符。
属性	See 资源属性.
属性规则	一种符合 JSON (p. 96) 的标记标准，用于声明 Amazon CloudFormation (p. 76) 模板中的属性、映射和输出值。
预置 IOPS	一个存储选项，可用于提供快速、可预测和一致的 I/O 性能。如果在创建数据库实例时指定 IOPS 速率，Amazon RDS (p. 73) 会为数据库实例的生命周期配置该 IOPS 速率。
虚拟参数	一种预定义设置（如 `AWS:StackName`），不必声明即可在 Amazon CloudFormation (p. 76) 模板中使用。在可使用常规参数的任何位置，都可以使用虚拟参数。
公用 AMI	所有 Amazon Web Services 账户 (p. 68)均有权启动的 Amazon Machine Image（AMI） (p. 72)。
公有数据集	一种大型公有信息集，可无缝整合到 Amazon Web Services 云 中的应用程序。Amazon 免费为社群存储公有数据集，与其他 Amazon 服务类似，用户只需为其应用程序所使用的计算和存储支付费用。这些数据集当前包括人类基因组计划、美国人口普查、Wikipedia 和其他来源的数据。 See Also https://aws.amazon.com/publicdatasets.
公有 IP 地址	一个公有数字地址（例如 192.0.2.44），供联网设备用通过 Internet 协议（IP）相互通信。每个 EC2 实例 (p. 89) 在启动时都被分配了两个 IP 地址，即私有地址（遵循 RFC 1918）和公有地址，它们可通过网络地址转换（NAT (p. 100)）直接相互映射。例外：在 Amazon VPC (p. 74) 中启动的实例只分配有私有 IP 地址。
公有子网	可从 Internet 访问其实例的subnet (p. 112)。
半虚拟化	半虚拟化。允许访客虚拟机在没有特殊支持扩展来实现完整硬件和 CPU 虚拟化的主机系统上运行。由于半虚拟化客户机运行未使用硬件模拟且经修改的操作系统，因此无法提供硬件相关的功能（如增强网络或 GPU 支持）。 See Also HVM 虚拟化.

Q

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) |

O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

四分位数分箱转换	Amazon Machine Learning：采用两种输入（一个数值变量和一个称作“分箱号”的参数）并输出类别变量的过程。四分位数分箱转换使机器学习模型能够理解数值变量分发的各个部分的单独重要性值，来发现变量分发中的非线性特征。
查询	一种 Web 服务，通常仅在 URL 中使用 GET 或 POST HTTP 方法和一个带有参数的查询字符串。 See Also REST.
查询字符串身份验证	一项 Amazon 功能，用于在 HTTP 请求查询字符串中而不是在`Authorization`标头中放置身份验证信息，从而支持对桶 (p. 83)中的对象进行基于 URL 的访问。
queue	保存在临时存储中等待传输或处理的消息或任务的序列。
队列 URL	用于唯一标识序列的 Web 地址。
配额	Amazon Web Services 账户 中资源、操作和项目的最大值。

R

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) | O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

range GET	用于指定下载操作要获取的数据字节范围的请求。如果对象很大，您可以通过发送多个范围 GET 请求（每个范围 GET 指定不同的 GET 字节范围）将下载操作分为若干较小的单位操作。
原始电子邮件	一种 sendmail 请求，可用于指定电子邮件标头和 MIME 类型。
RDS	See Amazon Relational Database Service (Amazon RDS).
只读副本	Amazon RDS (p. 73)：另一数据库实例的活动副本。源数据库实例上的任意数据更新都会利用 MySQL 5.1 内置的复制功能复制到只读副本数据库实例。
实时预测	Amazon Machine Learning：同步生成的单个数据观察的预测。 See Also 批量预测.
recipe	Amazon Personalize (p. 72)：一种预先配置的 Amazon Personalize 算法，用于预测用户将与之进行交互的项目（对于 USER_PERSONALIZATION 配方），或计算与用户表示感兴趣的特定项目类似的项目（对于 RELATED_ITEMS 配方），或您根据特定用户的预测兴趣对您提供的一系列项目进行排名（对于 PERSONALIZED_RANKING 配方）。 See Also USER_PERSONALIZATION 配方, RELATED_ITEMS 配方, PERSONALIZED_RANKING recipes.
推荐	Amazon Personalize (p. 72)：Amazon Personalize 预测用户将与之交互的项目列表。根据所使用的 Amazon Personalize 配方，推荐可以是物品列表（使用 USER_PERSONALIZATION 配方和 RELATED_ITEMS 配方），也可以是您提供的一系列物品的排名（使用 PERSONALIZED_RANKING 配方）。 See Also recipe, 活动, 解决方案版本, USER_PERSONALIZATION 配方, RELATED_ITEMS 配方, PERSONALIZED_RANKING recipes.
接收句柄	Amazon SQS (p. 73)：从队列接收消息时获得的标识符。从队列中删除消息或对消息的可见性超时进行更改时，您必须提供该标识符。
接收方	包含用于管理 收件人 (p. 106) 电子邮件传输操作的网络系统、软件和策略的实体。

收件人	Amazon Simple Email Service (Amazon SES) (p. 73)：接收电子邮件的人或实体。例如，某电子邮件里“收件人”字段中提及的人。
Redis	一种快速、开源、内存中的键-值数据结构存储。Redis 附带一组通用的内存中数据结构，您可以用来轻松创建各种自定义应用程序。
引用	一种将属性从一个Amazon 资源 (p. 106)插入到另一个 Amazon 资源中的方法。例如，您可以将 Amazon EC2 (p. 70) 安全组 (p. 109)属性插入到 Amazon RDS (p. 73) 资源。
区域	同一地理区域中的命名 Amazon 资源 (p. 106)集。一个区域至少由两个可用区 (p. 76)组成。
回归模型	Amazon Machine Learning：优化机器学习性能的常见数据转换的预先格式化指令。
回归模型	一种预测数值（例如，房屋的准确购买价格）的机器学习模型。
正规化	一个机器学习 (ML) 参数，可调整此参数来获得高质量 ML 模型。正规化有助于防止 ML 模型记住训练数据示例而不是了解如何泛化发现的模式（称作过度拟合）。在过度拟合训练数据时，机器学习（ML）模型非常适合训练数据，但不是很适合评估数据或新数据。
RELATED_ITEMS 配方	Amazon Personalize (p. 72)：推荐类似于指定物品的物品的配方，例如物品间（SIMS）配方。 See Also recipe, 物品间相似度 (SIMS) 配方.
替换环境	在 CodeDeploy 蓝/绿部署后的部署组中的实例。
副本分区	See 分片.
回复路径	接收回复电子邮件的电子邮件地址。它不同于退回路径 (p. 107)。
表述性状态转移	See REST.
声誉	1. 一种 Amazon SES (p. 73) 指标，涉及客户是否发送了高质量电子邮件，其依据为可能包含退回邮件 (p. 83)、投诉 (p. 84)和其他指标的因素。 2. 一个衡量置信度的指标，按照 互联网服务提供商（ISP） (p. 95) 或从其接收电子邮件的 IP 地址不为垃圾电子邮件 (p. 111)源的其他实体判断。
请求者	向 Amazon 发送请求以执行特定操作的人员（或应用程序）。Amazon 收到请求时，它首先评估请求者的权限以确定是否允许请求者执行请求操作（如果适用，对于请求的资源 (p. 106)）。
申请方付款	一种 Amazon S3 (p. 74) 功能，允许存储桶拥有者 (p. 83)指定请求访问特定桶 (p. 83)中对象的任何人必须支付数据传输和请求费用。
reservation	作为相同启动请求一部分启动的 EC2 实例 (p. 89)集合。不要与 Reserved Instance (p. 106) 混淆。
Reserved Instance	EC2 实例 (p. 89)的一个定价选项，可对满足指定参数的实例的按需 (p. 101)使用费用提供优惠。客户对实例的整个期限付款，无论客户使用实例的情况如何。
预留实例 Marketplace	一种在线交易，可将想要出售不再需要的预留容量的卖方与正在寻找购买额外容量的买方匹配起来。从第三方卖方购买的预留容量 (p. 106)的剩余期限短于完整的标准期限且可以按照不同的预付费用价格出售。使用费或经常性费用仍然相同，因为这些费用在购买预留实例时就已经设定。从 Amazon 获得的完全标准期限预留实例运行一年或三年。
资源	用户可以在 Amazon 中使用的实体，例如 EC2 实例 (p. 89)、Amazon DynamoDB (p. 70) 表、Amazon S3 (p. 74) 桶 (p. 83)、IAM (p. 78) 用户或 Amazon OpsWorks (p. 79) 堆栈 (p. 111)。

资源属性
在 Amazon CloudFormation (p. 76) 堆栈 (p. 111)中包含 Amazon 资源 (p. 106)时所需的值。每个资源都可以有一个或多个属性与之关联。例如，一个 `AWS::EC2::Instance` 资源可有一个 `UserData` 属性。即使资源没有属性，也必须在 Amazon CloudFormation 模板中声明一个属性部分。

资源记录
也称作资源记录集。域名系统 (DNS) 中的基本信息元素。
See Also 域名系统 on Wikipedia.

REST
表述性状态转移。一种简单的无状态架构，通常通过 HTTPS/TLS 运行。REST 强调，资源具有唯一分层标识符（URI），用常见媒体类型（例如 HTML、XML 或 JSON (p. 96)）表示，对资源的操作可以是预定义的，也可以是在媒体类型中发现的。实际上，这通常导致操作数量有限。
See Also 查询, WSDL, SOAP.

RESTful Web 服务
也称为 RESTful API。一种遵循 REST (p. 107) 架构限制的 Web 服务。API 操作必须明确使用 HTTP 方法，公布分层 URI 并传输 XML、JSON (p. 96) 或这二者。

启用返回
Amazon CloudSearch (p. 69)：一个在搜索结果中返回字段值的索引字段选项。

退回路径
发生退信时，将电子邮件回退到的地址。退回路径在原始电子邮件标头中指定。它不同于回复路径 (p. 106)。

修订
Amazon CodePipeline (p. 77)：对源操作中配置的源进行的更改，例如，将提交推送到 GitHub (p. 93) 存储库或更新受版本控制的 Amazon S3 (p. 74) 桶 (p. 83) 中的文件。

role
一个工具，用于授予对 Amazon Web Services 账户 (p. 68) 中 Amazon 资源 (p. 106)的临时访问权限。

回滚
退回至创建对象失败之前的状态，例如 Amazon CloudFormation (p. 76) 堆栈 (p. 111)。与故障相关的所有资源 (p. 106)在回滚期间会被删除。对于 Amazon CloudFormation，您可以在命令行上使用 `--disable-rollback` 选项来覆盖此行为。

根
Amazon Organizations (p. 80)：您的组织中的账户的父容器。如果您将 服务控制策略 (p. 109) 应用于根，它将应用于组织中的每个 组织单位 (p. 101) 和账户。

根凭证
与 Amazon Web Services 账户 (p. 68) 拥有者关联的身份验证信息。

根设备卷
一个 volume (p. 116)，其中包含用于启动 实例 (p. 95) 的映像 (也称为根设备)。如果您从由实例存储 (p. 95)支持的 AMI (p. 72) 启动实例，则为从存储在 Amazon S3 (p. 74) 中的模板创建的实例存储volume (p. 116)。如果您从由 Amazon EBS (p. 70) 支持的 AMI 启动实例，则为从 Amazon EBS 快照创建的 Amazon EBS 卷。

路由表
一组路由规则，用于对离开与路由表关联的任意subnet (p. 112)的流量进行控制。您可以为同一个路由表关联多个子网，但是只能为一个子网关联一个路由表。

行标识符
Amazon Machine Learning：输入数据中的一个属性，可在评估输出或预测输出中包含此属性以便更轻松地将预测与观察关联。

规则
Amazon WAF (p. 82)：Amazon WAF 在针对 Amazon 资源 (p. 106)（例如 Amazon CloudFront (p. 69) 分配）的 Web 请求中搜索的一组条件。您将规则添加到 Web ACL (p. 116)，然后根据每个规则指定是要允许还是阻止 Web 请求。

S

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) |

O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

S3	See Amazon Simple Storage Service (Amazon S3).
采样周期	明确定义的持续时间（如：一分钟），Amazon CloudWatch (p. 69) 每隔此时间计算一次统计数据 (p. 111)。
沙盒	供您进行应用程序功能测试的测试位置（不会影响生产、产生费用或购买产品）。 Amazon SES (p. 73)：可供开发人员用于测试和评估服务的环境。在沙盒中，您拥有 Amazon SES API 的所有访问权限，但只能向经过验证的电子邮件地址和邮箱模拟器发送电子邮件。要离开沙盒，您必须申请生产环境访问权限。沙盒中账户的发送限制 (p. 109) 要低于生产账户。
缩减	从 自动扩缩组 (p. 75)中删除 EC2 实例。
扩展	将 EC2 实例添加到 自动扩缩组 (p. 75)。
扩展策略	关于自动扩缩组应如何自动扩缩 自动扩缩组 (p. 75) 以响应不断变化的需求的描述。 See Also 缩减, 扩展.
扩展活动	一种通过启动或终止实例来更改 自动扩缩组 (p. 75) 大小、配置或部署的进程。
计划程序	用于将任务 (p. 113)放置在容器实例 (p. 85)中的方法。
schema	Amazon Machine Learning：解释机器学习模型的输入数据时所需的信息，包括属性名称及其分配的数据类型和特殊属性名称。
分数截断值	Amazon Machine Learning：二进制分类模型输出一个介于 0 和 1 之间的分数。要确定将观察分类为 1 还是 0，您可以选取分类阈值或截断值，Amazon ML 会将它与分数进行比较。当目标等于 1 时，将预测分数高于截断值的观察；当目标等于 0 时，将预测分数低于截断值的观察。
SCP	See 服务控制策略.
搜索 API	Amazon CloudSearch (p. 69)：用于向搜索域 (p. 108)提交搜索请求的 API。
搜索域	Amazon CloudSearch (p. 69)：封装了可搜索数据和用于处理搜索请求的搜索实例。您通常需要为每个不同的待搜索数据集合设置单独的 Amazon CloudSearch 域。
搜索域配置	Amazon CloudSearch (p. 69)：域的索引选项、分析方案 (p. 74)、表达式 (p. 91)、建议索引 (p. 112)、访问策略以及扩展和可用性选项。
启用搜索	Amazon CloudSearch (p. 69)：一个使字段数据可被搜索的索引字段选项。
搜索终端节点	Amazon CloudSearch (p. 69)：向搜索域发送搜索请求时所连接的 URL。每个 Amazon CloudSearch 域都拥有唯一的搜索终端节点，该终端节点在该域的生命周期内保持不变。
搜索索引	Amazon CloudSearch (p. 69)：一种便于快速、准确数据检索的可搜索数据表示形式。
搜索实例	Amazon CloudSearch (p. 69)：用于索引数据和处理搜索请求的计算资源 (p. 106)。一个 Amazon CloudSearch 域有一个或多个搜索实例，每个实例具有一定量的 RAM 和 CPU 资源。随着数据量的增长，将部署更多的搜索实例或更大的搜索实例，以包含索引数据。如有必要，索引会自动在多个搜索实例间分区。随着请求量或复杂度的增加，每个搜索分区会自动复制，以提供额外的处理容量。

搜索请求	Amazon CloudSearch (p. 69)：发送到 Amazon CloudSearch 域的搜索终端节点以便从与特定搜索条件匹配的索引检索文档的请求。
搜索结果	Amazon CloudSearch (p. 69)：与搜索请求匹配的文档。也称作搜索命中结果。
私有访问密钥	与 访问密钥 ID (p. 67) 一起用于对编程方式的 Amazon 请求进行加密签名的密钥。对请求进行签名可标识发送方，并防止请求被修改。您可以为您的 Amazon Web Services 账户 (p. 68)、单个 IAM 用户 (p. 114)和临时会话生成秘密访问密钥。
安全组	实例允许的一组命名入站网络连接。（Amazon VPC (p. 74) 中的安全组还包括出站连接支持。）每个安全组都包含协议、端口和 IP 地址范围的列表。安全组可应用于多个实例，多个组可控制同一个实例。
发件人	发送电子邮件的人或实体。
发件人 ID	SPF (p. 111) 的 Microsoft 控制版本。电子邮件身份验证和反欺诈系统。有关发件人 ID 的更多信息，请参阅 Wikipedia 中的发件人 ID。
发送限制	与每个 Amazon SES (p. 73) 账户关联的发送配额 (p. 109)和最大发送速率 (p. 99)。
发送配额	您在 24 小时内可以使用 Amazon SES (p. 73) 发送的电子邮件的最大数目。
服务器端加密 (SSE)	在服务器级别对数据进行加密 (p. 90)。Amazon S3 (p. 74) 支持三种服务器端加密模式：SSE-S3（其中，Amazon S3 管理密钥）、SSE-C（其中，客户管理密钥）和 SSE-KMS（其中，Amazon Key Management Service (Amazon KMS) (p. 79) 管理密钥）。
服务控制策略	Amazon Organizations (p. 80)：一种基于策略的控制，可指定用户和角色可在服务控制策略 (SCP) 影响的账户中使用的服务和操作。
服务终端节点	See endpoint.
服务运行状况控制面板	显示有关 Amazon 服务可用性实时信息（分钟级）的网页。此控制面板位于 http://status.aws.amazon.com/。
服务限额	一项服务，可随着 Amazon 工作负载的增长轻松、大规模地查看和管理配额。配额也称为限制，是您可以在 Amazon Web Services 账户 中创建的最大资源数。
服务角色	一个 IAM (p. 78) role (p. 107)，可用于向 Amazon 服务授予权限以使其能够访问 Amazon 资源 (p. 106)。附加到服务角色的策略将确定服务可访问的 Amazon 资源以及可使用这些资源执行的操作。
SES	See Amazon Simple Email Service (Amazon SES).
session	Amazon Security Token Service (Amazon STS) (p. 81) 提供的临时安全凭证允许访问 Amazon Web Services 账户 的期间。
SHA	安全哈希算法。SHA1 是早期版本的算法，Amazon 已将其替换为 SHA256。
分片	Amazon OpenSearch Service (OpenSearch Service) (p. 71)：索引中的数据分区。可以将一个索引拆分成多个分片，它们可包括主分片（原始分片）和副本分片（主分片的副本）。副本分片提供失效转移功能。这意味着，如果包含主分片的集群节点失败，副本分片会被提升为主分片。副本分片也可以处理请求。
共享 AMI	开发人员构建和提供给他人使用的 Amazon Machine Image（AMI） (p. 72)。
关闭操作	Amazon EMR (p. 71)：启动脚本（在终止任务流之前并行执行一系列命令）的预定义引导操作。

签名	指的是数字签名，这是一种确认数字信息真实性的数学方式。Amazon 使用签名验证您发送到我们 Web 服务的请求。有关更多信息，请参阅 https://aws.amazon.com/security。
签名文件	Amazon Import/Export (p. 78)：您复制到存储设备根目录中的文件。该文件包含任务 ID、清单文件和签名。
Signature Version 4	用于验证针对所有 Amazon Web Services 区域 中的 Amazon 服务的入站 API 请求的协议。
简单邮件传输协议	See SMTP.
简单对象访问协议	See SOAP.
Simple Storage Service	See Amazon Simple Storage Service (Amazon S3).
SIMS 配方	See 物品间相似度 (SIMS) 配方.
单点登录	一种身份验证方案，允许用户登录一次，以访问多个应用程序和网站。Amazon Single Sign-On 服务名称现在为 Amazon IAM Identity Center (successor to Amazon Single Sign-On)。 See Also Amazon IAM Identity Center (successor to Amazon Single Sign-On).
单可用区数据库实例	部署在一个可用区： (p. 76)中的标准（非多可用区）数据库实例 (p. 87)，在另一个可用区中没有备用副本。 See Also 多可用区部署.
模糊短语搜索	指定词汇必须相互有多接近才视为匹配的短语搜索。
SMTP	简单邮件传输协议。用来在互联网主机之间交换电子邮件以便进行路由和传送的标准。
snapshot	Amazon Elastic Block Store (Amazon EBS) (p. 70)：卷 (p. 116)的备份，存储在 Amazon S3 (p. 74) 中。您可以将这些快照用作新 Amazon EBS 卷的起点，或使用这些快照保护您的数据以实现长期持久性。 See Also 数据库快照.
SNS	See Amazon Simple Notification Service (Amazon SNS).
SOAP	简单对象访问协议。一个基于 XML 的协议，可用于通过特定协议（例如 HTTP 或 SMTP）在应用程序之间交换信息。 See Also REST, WSDL.
软退回邮件	一种临时电子邮件发送失败情况，例如整个邮箱出现故障。
软件 VPN	通过 Internet 实现的基于软件设备的 VPN 连接。
solution	Amazon Personalize (p. 72)：可用于生成推荐的配方、自定义参数和经过训练的模型（解决方案版本）。 See Also recipe, 解决方案版本, 推荐.
解决方案版本	Amazon Personalize (p. 72)：作为 Amazon Personalize 解决方案的一部分创建的经过训练的模型。您可以在活动中部署解决方案版本以生成推荐。 See Also solution, 活动, 推荐.
启用排序	Amazon CloudSearch (p. 69)：一个索引字段选项，该选项支持使用一个字段对搜索结果进行排序。
排序键	一个用于对复合主键中的分区键进行排序的属性（也称作范围属性）。 See Also 分区键, 主键.

源/目标检查	一种安全措施，用于验证 EC2 实例 (p. 89) 是否是它所发送的所有流量的源，以及它接收的所有流量的最终目标。换句话说，此措施用于验证实例是否不中继流量。默认情况下，会启用源/目标检查。对于用作网关的实例，如 VPC (p. 116) NAT (p. 100) 实例，必须禁用源/目标检查。
垃圾电子邮件	未经请求的批量电子邮件。
spamtrap	由 anti-垃圾电子邮件 (p. 111) 实体设置的电子邮件地址。此电子邮件地址不用于通信，而是监控未经请求的电子邮件。它也称为蜜罐。
SPF	发件人策略框架。电子邮件验证标准。
Spot 实例	一种 EC2 实例 (p. 89)，您可以出价以利用未使用的 Amazon EC2 (p. 70) 容量。
Spot 价格	在任何指定时间 Spot 实例 (p. 111) 的价格。如果您的最高价高于当前价格，并且满足您的限制，则 Amazon EC2 (p. 70) 会代表您启动实例。
SQL 注入匹配条件	Amazon WAF (p. 82)：一个属性，指定 Amazon WAF 从中检查恶意 SQL 代码的 Web 请求部分（如标头或查询字符串）。根据指定的条件，可以将 Amazon WAF 配置为允许或阻止针对 Amazon 资源 (p. 106)（如 Amazon CloudFront (p. 69) 分发）的 Web 请求。
SQS	See Amazon Simple Queue Service (Amazon SQS).
SSE	See 服务器端加密 (SSE).
SSL	安全套接字层 See Also 传输层安全性 (TLS).
堆栈	Amazon CloudFormation (p. 76)：作为一个整体创建和删除的 Amazon 资源的集合。 Amazon OpsWorks (p. 79)：您集中管理的一组实例，通常，它们具有共同的用途，例如为 PHP 应用程序提供服务。堆栈作为容器使用，用于整体性处理适用于实例组的任务，例如管理应用程序和食谱。
站	Amazon CodePipeline (p. 77)：可从中执行一个或多个操作的管道工作流的部分。
站	Amazon设施上的一个位置，您的 Amazon Import/Export 数据从该位置传输到您的存储设备或从该设备传出到该位置。
统计数据	在给定 采样周期 (p. 108) 内提交的值的五个函数之一。这些函数是 `Maximum`、`Minimum`、`Sum`、`Average` 和 `SampleCount`。
词干	一组相关词共同的通用根或子字符串。
提取词干	将相关词映射到公共词干的过程。该过程支持使一个词的不同变体匹配。例如，搜索“horse”可能将“horses”、“horseback”和“horsing”以及“horse”作为匹配项返回。Amazon CloudSearch (p. 69) 支持基于词典和通过算法来提取词根。
step	Amazon EMR (p. 71)：在任务流程 (p. 96)中应用于数据的单个函数。所有步骤的总和组成任务流程。
步骤类型	Amazon EMR (p. 71)：在一个步骤内完成的工作的类型。步骤类型的数量有限，例如将数据从 Amazon S3 (p. 74) 移至 Amazon EC2 (p. 70) 或从 Amazon EC2 移至 Amazon S3。
粘性会话	弹性负载均衡 (p. 90) 负载均衡器的一个功能，可用于将用户的会话绑定到特定应用程序实例。这可在会话期间将来自用户的所有请求发送到相同的应用程序实例中。但负载均衡器默认情况下将每个请求单独路由到负载最小的应用程序实例。

stopping	从索引或搜索请求中筛选非索引字的过程。
非索引字	未编制索引的字词，会自动从搜索请求中筛除，因为它不重要或太常见，将它包括进去会产生过多匹配项，从而导致搜索结果中出现太多无用项。非索引字是语言特定的。
流式处理	Amazon EMR (p. 71)：Hadoop (p. 93) 附带的实用工具，可用于使用非 Java 语言开发 MapReduce 可执行文件。 Amazon CloudFront (p. 69)：实时使用媒体文件的能力（这种文件从服务器以稳定流的形式传输）。
串流分配	使用实时消息传输协议 (RTMP) 连接提供流媒体文件的一种特殊类型的 分配 (p. 88)。
流	See Amazon Kinesis Data Streams.
待签字符串	在计算 HMAC (p. 94) 签名之前，首先要按规范顺序汇编必需的组件。预先加密的字符串是要签名的字符串。
字符串匹配条件	Amazon WAF (p. 82)：一个属性，指定 Amazon WAF 在 Web 请求中搜索的字符串（如标头中的值或查询字符串）。根据指定的字符串，可以将 Amazon WAF 配置为允许或阻止针对 Amazon 资源 (p. 106)（如 CloudFront (p. 69) 分发）的 Web 请求。
强一致性读取	一个读取过程，其会返回具有最新数据的响应。此数据反映来自所有之前已成功写入操作的更新，与区域无关。 See Also 数据一致性, 最终一致性, 最终一致性读取.
结构化查询	使用 Amazon CloudSearch (p. 69) 结构化查询语言指定的搜索条件。通过使用结构化查询语言，可以构造使用高级搜索选项并使用布尔运算符合并多个搜索条件的复合查询。
STS	See Amazon Security Token Service (Amazon STS).
subnet	EC2 实例 (p. 89) 可挂载到的 VPC (p. 116) 的一段 IP 地址范围。您可以根据安全性和运营需求创建子网对实例分组。
订阅按钮	一个 HTML 编码的按钮，可用于轻松向客户收取定期费用。
建议索引	Amazon CloudSearch (p. 69)：指定一个索引字段，用于获取自动填写建议和选项，以便可以启用模糊匹配并控制建议的排序方式。
建议	包含 建议索引 (p. 112) 指定的字段中部分搜索字符串的匹配项的文档。Amazon CloudSearch (p. 69) 建议包含文档 ID 和每个匹配文档的字段值。要成为匹配项，字符串必须从字段开头与字段内容匹配。
支持的 AMI	Amazon Machine Image（AMI） (p. 72) 类似于 付费 AMI (p. 102)，不同之处是拥有者针对客户在其自己的 AMI 中使用的其他软件或服务收费。
SWF	See Amazon Simple Workflow Service (Amazon SWF).
对称加密	仅使用私有密钥的加密 (p. 90)。 See Also 非对称加密.
同步退回邮件	一种 退回邮件 (p. 83)，当 发件人 (p. 109) 和 接收方 (p. 105) 的电子邮件服务器正在通信时发生。
同义词	一个与索引字相同或接近、在搜索请求中指定时可能产生相同结果的字词。例如，搜索“Rocky Four”或“Rocky 4”可能返回第四部 Rocky 电影。您可以通过指定 `four` 和 `4` 是 `IV` 的同义词来实现。同义词是语言特定的。

T

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) | O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

table
: 数据集合。类似于其他数据库系统，DynamoDB 将数据存储在表中。

标签
: 您可以定义并分配到 Amazon 资源 (p. 106)（如 EC2 实例 (p. 89)）的元数据。并非所有 Amazon 资源都可添加标签。

标记
: 标记资源：将标签 (p. 113)应用于 Amazon 资源 (p. 106)。

  Amazon SES (p. 73)：也称为贴标签。一种设置 退回路径 (p. 107) 电子邮件地址格式以便您为每个邮件收件人指定不同退回路径的方式。您可以使用标记支持 VERP (p. 115)。例如，如果 Andrew 管理一个邮件列表，他可以使用退回路径 `andrew+recipient1@example.net` 和 `andrew+recipient2@example.net`，这样他可以确定哪一封电子邮件被退回。

目标属性
: Amazon Machine Learning (Amazon ML)：输入数据中包含“正确”答案的属性。Amazon ML 使用目标属性了解如何预测新数据。例如，如果您构建用于预测房屋销售价格的模型，则目标属性将为“目标销售价格（美元）”。

目标版本
: Amazon CodeDeploy (p. 77)：已上传到存储库并且将部署到部署组中的实例的应用程序修订的最新版本。换言之，当前面向部署的应用程序版本。这也是将为自动部署提取的修订。

task
: 在容器实例 (p. 85)上运行的任务定义 (p. 113)的实例化。

任务定义
: 任务的蓝图。指定 task (p. 113) 名称、修订、容器定义 (p. 85)和 volume (p. 116) 信息。

任务节点
: 一个 EC2 实例 (p. 89)，运行 Hadoop (p. 93) 映射和缩减任务，但不存储数据。任务节点由 主节点 (p. 99) 管理，后者将 Hadoop 任务分配到节点并监控它们的状态。在任务流程运行期间，您可以增加或减少任务节点的数量。因为它们不存储数据，可以从任务流程中添加和移除，所以您可以使用任务节点管理您的任务流程使用的 EC2 实例容量，增加容量以处理峰值负载，并在之后减少容量。

  任务节点仅运行 TaskTracker Hadoop 守护程序。

tebibyte (TiB)
: 万亿二进制字节的缩写形式。1 太字节（TiB）为 2^40 字节，即 1.099511627776 万亿字节。1 terabyte (TB) 为 10^12 字节，即 1000000000000 字节。1024 TiB 为 1 pebibyte (PiB) (p. 102)。

模板格式版本
: Amazon CloudFormation (p. 76) 模板设计的版本，用于确定可用功能。如果模板中省略了 `AWSTemplateFormatVersion` 部分，则 Amazon CloudFormation 将采用最新的格式版本。

模板验证
: 确认在 Amazon CloudFormation (p. 76) 模板中使用 JSON (p. 96) 代码的过程。您可以使用 `cfn-validate-template` 命令验证所有 Amazon CloudFormation 模板。

临时安全凭证
: Amazon STS (p. 81) 在您调用 STS API 操作时提供的身份验证信息。包括访问密钥 ID (p. 67)、私有访问密钥 (p. 109)、session (p. 109)标记和过期时间。

限制
: 根据一个或多个限制来自动限制或减慢过程。例如，如果某个应用程序（或对相同流执行操作的一组应用程序）尝试以高于分片限制的速率从分片中获取数据，则 Amazon Kinesis Data Streams (p. 71) 将节流操作。Amazon

	API Gateway (p. 68) 使用节流来限制单个账户的稳态请求速率。Amazon SES (p. 73) 使用限制拒绝尝试发送超出发送限制 (p. 109)的电子邮件的操作。
时间序列数据	作为指标的一部分提供的数据。时间值为当该值出现时假定的值。指标是 Amazon CloudWatch (p. 69) 的基本概念，代表了数据点的时间顺序集。您可以将指标数据点发布到 CloudWatch 中，稍后能够以时间序列有序数据集的形式检索关于这些数据点的统计数据。
timestamp	ISO 8601 格式的日期/时间字符串（具体来说，格式为 `YYYY-MM-DD`）。
TLS	See 传输层安全性 (TLS).
令牌化	通过可以检测的边界（如空格和连字符）将文本流分割到不同令牌中的过程。
topic	发送消息和订阅通知的通信渠道。它为发布者和用户相互交流提供了一个接入点。
流量镜像	一项 Amazon VPC 功能，您可以使用该功能复制 Amazon EC2 实例的弹性网络接口中的网络流量。然后将该网络流量发送到带外安全和监控设备来进行内容检查、威胁监控和故障排除。 See Also https://aws.amazon.com/vpc/.
训练数据源	一个数据源，其中包含 Amazon Machine Learning 用于训练机器学习模型以做出预测的数据。
转换	Amazon CodePipeline (p. 77)：从工作流中的一个阶段继续执行到另一个阶段的管道修订操作。
传输层安全性 (TLS)	对通过 Internet 的通信提供安全性的加密协议。它的前身是安全套接字层 (SSL)。
信任策略	IAM (p. 78) 策略 (p. 103)，是 IAM role (p. 107)的固有组成部分。信任策略指定允许哪些委托人使用角色。
可信密钥组	Amazon CloudFront 密钥组，CloudFront 可以使用这些密钥组中的公有密钥验证 CloudFront 签名 URL 和签名 Cookie 的签名。
可信的签名者	请参阅可信密钥组 (p. 114)。
优化	选择 AMI (p. 72) 的数量和类型，以最高效地运行 任务流程。Hadoop (p. 93)
隧道	使用 Internet 来连接私有网络节点的私有网络流量传输路由。隧道使用加密和安全协议（如 PPTP）来防止流量在经过公有路由节点时被拦截。

U

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) | O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

无界	潜在事件发生的次数不受设定次数的限制。在 WSDL (p. 116) 中，在定义列表型数据类型（例如 `maxOccurs="unbounded"`）时经常使用此值。
单位	作为指标数据提交给 Amazon CloudWatch (p. 69) 的值的标准衡量指标。单位包括“秒”、“百分比”、“字节”、“比特”、“计数”、“字节/秒”、“比特/秒”、“计数/秒”和“无”。
使用率报告	一个详述对特定 Amazon 服务的使用情况的 Amazon 记录。您可以从 https://aws.amazon.com/usage-reports/ 生成和下载使用情况报告。
user	对 Amazon 产品进行 API 调用的 账户 (p. 68) 下的人员或应用程序。在 Amazon Web Services 账户 内，每个用户均有唯一的用户名和一组无法与其他用户共享的安

	全凭证。这些凭证独立于 Amazon Web Services 账户 的安全凭证。每个用户均与 Amazon Web Services 账户 且仅与一个账户关联。
用户数据集	Amazon Personalize (p. 72)：用于保存用户相关元数据（例如年龄、性别或会员资格）的容器。 See Also dataset.
用户-个性化配方	Amazon Personalize (p. 72)：一个基于 HRNN 的 USER_PERSONALIZATION 食谱，可用于预测用户与之交互的项目。用户-个性化配方可以使用物品浏览和展示数据生成有关新物品的推荐。 See Also HRNN, recipe, USER_PERSONALIZATION 配方, 物品浏览, 展示数据, 推荐.
USER_PERSONALIZATION 配方	Amazon Personalize (p. 72)：用于构建推荐系统的配方，该系统根据交互数据集、项目数据集和用户数据集中提供的数据预测用户与之交互的项目。 See Also recipe, 用户-个性化配方, 热门程度-计数配方, HRNN.

V

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) | O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

验证	See 模板验证.
值	项目属性 (p. 75)的实例，如电子表格中的单元格。一个属性可以有多个值。 标记资源：一个特定的标签 (p. 113)标签，可充当标记类别（密钥）中的描述符。例如，您可能有一个标签键为 Owner 和标签值为 Jan 的 EC2 实例 (p. 89)。您最多可以使用 10 个键值对标记 Amazon 资源 (p. 106)。并非所有 Amazon 资源都可添加标签。
可变信封退回路径	See VERP.
验证	确认您具有用于收发电子邮件的电子邮件地址或域的过程。
VERP	可变信封退回路径。电子邮件发送应用程序可将退回 (p. 83)邮件与无法送达的地址（因为对每个收件人使用不同的 退回路径 (p. 107) 而导致退回邮件）进行匹配的方法。VERP 通常用于邮件列表。借助 VERP，收件人的电子邮件地址将嵌入到退回路径的地址中，此地址是退回的邮件所返回的位置。这样，可以自动处理所退回电子邮件，而无需打开退回邮件（可能在内容上有所不同）。
版本控制	Amazon S3 (p. 74) 中的每个对象都有一个键和一个版本 ID。密钥相同但版本 ID 不同的对象可存储在同一桶 (p. 83)中。版本控制是在存储段层使用 PUT Bucket 版本控制功能启用的。
VGW	See 虚拟私有网关（VGW）.
虚拟化	允许多个客户虚拟机 (VM) 在主机操作系统运行。根据虚拟化类型，客户虚拟机可在主机硬件上的一个或多个级别上运行。 See Also 半虚拟化, HVM 虚拟化.
Virtual Private Cloud	See VPC.
虚拟私有网关（VGW）	维护连接的 VPN 连接 (p. 116)的 Amazon 端。虚拟专用网关的内部接口通过 VPN 附件连接到您的 VPC (p. 116)。外部接口连接到 VPN 连接，这会引向客户网关 (p. 86)。

可见性超时	一段时间，在应用程序组件从队列中获取消息后，这段时间内该消息对应用程序其余部分不可见。在可见性超时期间，接收到消息的组件通常先处理消息，然后将其从队列中删除。这会防止多个组件处理同一消息。
VM Import/Export	一项服务，用于将虚拟机（VM）镜像从现有的虚拟化环境导入 Amazon EC2，然后再导回。 See Also https://aws.amazon.com/ec2/vm-import.
volume	实例 (p. 95)上的固定存储量。当容器不再运行时，您可以在多个容器 (p. 85)之间共享卷数据并将数据保留在容器实例 (p. 85)上。
VPC	Virtual Private Cloud。一个由具有共同的安全性和互连性的基础设施、平台和应用程序服务填充的弹性网络。
VPC 终端节点	一项功能，可用于在 VPC (p. 116) 与其他 Amazon 服务之间创建私有连接，无需通过互联网、NAT (p. 100) 实例、VPN 连接 (p. 116)或 Amazon Direct Connect (p. 77) 进行访问。
VPG	See 虚拟私有网关（VGW）.
VPN CloudHub	See Amazon VPN CloudHub.
VPN 连接	Amazon Web Services (Amazon) (p. 74)：VPC (p. 116) 和其他某个网络（例如企业数据中心、家庭网络或主机托管设施）之间的 IPsec 连接。

W

Numbers and symbols (p. 67) | A (p. 67) | B (p. 82) | C (p. 83) | D (p. 86) | E (p. 89) | F (p. 92) | G (p. 93) | H (p. 93) | I (p. 94) | J (p. 96) | K (p. 96) | L (p. 97) | M (p. 98) | N (p. 100) | O (p. 101) | P (p. 102) | Q (p. 104) | R (p. 105) | S (p. 107) | T (p. 113) | U (p. 114) | V (p. 115) | W (p. 116) | X, Y, Z (p. 116)

WAM	See Amazon WorkSpaces Application Manager (Amazon WAM).
Web 访问控制列表（Web ACL）	Amazon WAF (p. 82)：一组规则，定义 Amazon WAF 在针对 Amazon 资源 (p. 106)（如 Amazon CloudFront (p. 69) 分发）的 Web 请求中搜索的条件。Web 访问控制列表（Web ACL）指定是允许、阻止请求还是对请求进行计数。
Web 服务描述语言	See WSDL.
WSDL	Web 服务描述语言。一种语言，用于描述 Web 服务可执行的操作以及操作请求和响应的语法。 See Also REST, SOAP.

X、Y、Z

X.509 证书	一个数字文档，使用 X.509 公有密钥基础设施（PKI）标准来验证公有密钥是否属于证书 (p. 84) 中描述的实体。
yobibyte (YiB)	亿亿亿二进制字节的缩写形式。1 尧字节（YiB）为 2^80 字节，即 1.2089258196146291747061760 亿亿亿字节。1 yottabyte (YB) 为 10^24 字节，即 1000000000000000000000000 字节。
zebibyte (ZiB)	十万亿亿二进制字节的缩写形式。1 泽字节（ZiB）为 2^70 字节，即 1.180591620717411303424 十万亿亿字节。1 zettabyte (ZB) 为 10^21 字节，即 1000000000000000000000 字节。1024 ZiB 为 1 yobibyte (YiB) (p. 116)。
区域感知	Amazon OpenSearch Service (OpenSearch Service) (p. 71)：一项配置，可在跨同一区域中两个可用区 (p. 76)的集群中分配节点。当节点和数据中心出现故

障时，区域感知可帮助防止数据丢失并最大程度地缩短停机时间。如果启用了区域感知，则在实例计数中必须有偶数数目的数据实例，并且还必须使用 Amazon OpenSearch Service 配置 API 为您的 OpenSearch 集群复制数据。