

亚马逊销售合作伙伴 **API** 开发者指南

内容

什么是 销售合作伙伴 API ?	4
销售合作伙伴 API HTTP 方法	4
销售合作伙伴 API 端点	4
marketplaceId 值	5
混合 销售合作伙伴 API 应用程序	6
全球应用程序	7
注册您的 销售合作伙伴 API 应用程序	8
第 1 步。创建 AWS 账户	8
第 2 步。创建 AWS IAM 策略	8
第 3 步。创建 IAM 用户	8
第 4 步。提供您的应用程序注册信息	9
查看您的开发者信息	11
授权销售合作伙伴 API 应用程序	12
商城应用程序商店工作流程	12
第 1 步。卖家从商城应用程序商店启动授权	12
第 2 步。卖家同意授权您的应用程序	13
第 3 步。卖家登录您的网站	13
第 4 步。亚马逊向您发送授权信息	14
第 5 步。您的应用程序用 LWA 授权码交换 LWA 刷新令牌	14
网站工作流程	15
第 0 步。设置 OAuth 授权 URI	16
第 1 步。卖家从您的网站启动授权	16
第 2 步。卖家同意授权您的应用程序	16
第 3 步。亚马逊向您发送授权信息	17
第 4 步。您的应用程序用 LWA 授权码交换 LWA 刷新令牌	17
自行授权	19
通过 LWA 令牌交换和身份验证生成 Java SDK	20
使用生成的 Java SDK 连接到销售合作伙伴 API	22
生成 Java 客户端库	24
连接到 销售合作伙伴 API	25
第 1 步。请求使用亚马逊账号登录访问令牌	25
第 2 步。构建 销售合作伙伴 API URI	26
第 3 步。将标头添加到 URI	26
第 4 步。创建并签署您的请求	27

3

亚马逊销售合作伙伴 API 开发者指南

3

凭据范围.....

Authorization 标头.....

28

28

响应格式.....

29

免授权操作.....

31

限制：针对提交请求频率的限制.....

32

在所有请求中包含 **User-Agent** 标头.....

33

销售合作伙伴 **API** 沙箱.....

34

如何对销售合作伙伴 **API** 进行沙箱调用.....

34

销售合作伙伴 **API** 沙箱端点.....

34

常见问题.....

36

销售合作伙伴 **API** 与亚马逊商城网络服务有何不同？

36

什么是 销售合作伙伴 API?

销售合作伙伴 API 是一个基于 REST 的 API，亚马逊卖家可以使用这些接口，以编程方式访问他们有关商品、订单、付款、报告等的信息。使用 销售合作伙伴 API 的应用程序可以提高销售效率、减少人工需求并缩短响应买家的时间，从而帮助卖家发展业务。销售合作伙伴 API 基于亚马逊商城网络服务（亚马逊 MWS）的功能而构建，但提供了一些功能，用来提高开发者以及与他们合作的卖家的可用性和安全性。

销售合作伙伴 API 测试版计划

销售合作伙伴 API 测试版计划仅限受邀者参与。需要亚马逊支持工程师的帮助，才能完成[注册销售合作伙伴 API 应用程序](#)等任务。您可以通过打开一个[支持实例](#)来就测试版计划接洽亚马逊支持工程师。请在支持实例的主题行中包含“销售合作伙伴 API 测试版”。

以下是自从 Alpha 计划以来对 销售合作伙伴 API 的更改：

- 您可以设置卖家从商城应用商店详情页面启动的 OAuth 授权工作流程。有关更多信息，请参阅[商城应用程序商店工作流程](#)。
- 您可以设置卖家从您自己的网站启动的 OAuth 授权工作流程。有关更多信息，请参阅[网站工作流程](#)。
- 您可以创建能同时调用销售合作伙伴 API 和亚马逊 MWS 的混合应用程序。有关更多信息，请参阅[纯粹和混合的销售合作伙伴 API 应用程序](#)。
- 我们已经为北美、欧洲和远东地区添加了新的端点。请参阅[销售合作伙伴 API 端点](#)。
- 我们支持多个新的市场。请参阅[marketplaceId 值](#)。

纯粹和混合的销售合作伙伴 API 应用程序

您可以构建两种类型的 销售合作伙伴 API 应用程序：

- 纯粹的销售合作伙伴 API 应用程序。仅调用销售合作伙伴 API。
- 混合的销售合作伙伴 API 应用程序。可同时调用 销售合作伙伴 API 和亚马逊 MWS。

有关更多信息，请参阅[混合销售合作伙伴 API 应用程序](#)。

销售合作伙伴 API HTTP 方法

销售合作伙伴 API 支持这些 HTTP 方法。

HTTP 方法	描述
GET	检索资源数据或资源列表。
POST	将实体提交到指定的资源，通常会导致状态更改或给服务器造成副作用。
PUT	将目标资源的所有当前表示形式替换为请求负载。

销售合作伙伴 API 端点

销售合作伙伴 API 端点与特定 AWS 区域相关联。AWS 区域非常重要，因为它是凭据范围的一部分，是在调用销售合作伙伴 API 时计算签名所必需的。有关更多信息，请参阅[凭据范围](#)。

销售区域	端点	AWS 区域
北美（加拿大、美国、墨西哥和巴西商城）	https://sellingpartnerapi-na.amazon.com	us-east-1

销售区域	端点	AWS 区域
欧洲（西班牙、英国、法国、德国、意大利、土耳其、阿联酋和印度商城）	https://sellingpartnerapi-eu.amazon.com	eu-west-1
远东（新加坡、澳大利亚和日本市场）	https://sellingpartnerapi-fe.amazon.com	us-west-2

marketplaceId 值

marketplaceId 标识请求的商城。

北美

国家/地区	marketplaceId	国家/地区代码
加拿大	A2EUQ1WTGCTBG2	CA
美国	ATVPDKIKX0DER	US
墨西哥	A1AM78C64UM0Y8	MX
巴西	A2Q3Y263D00KWC	BR

欧洲

国家/地区	marketplaceId	国家/地区代码
西班牙	A1RKKUPIHCS9HS	ES
英国	A1F83G8C2ARO7P	GB
法国	A13V1IB3VIYZZH	FR
德国	A1PA6795UKMFR9	DE
意大利	APJ6JRA9NG5V4	IT
土耳其	A33AVAJ2PDY3EV	TR
阿拉伯联合酋长国	A2VIGQ35RCS4UG	AE
印度	A21TJRUN4KGV	IN

远东地区

国家/地区	marketplaceId	国家/地区代码
新加坡	A19VAU5U5O7RUS	SG
澳大利亚	A39IBJ37TRP1C6	AU
日本	A1VC38T7YXB528	JP

混合 销售合作伙伴 API 应用程序

混合 销售合作伙伴 API 应用程序是一种可同时调用 销售合作伙伴 API 和亚马逊商城网络服务（亚马逊 MWS）的应用程序。当您的解决方案需要两种服务的功能时，请使用混合应用程序。当卖家授权您的混合 销售合作伙伴 API 应用程序时，他们将 (1) 授权您的亚马逊 MWS 开发者编号代表他们调用亚马逊 MWS，以及 (2) 授权应用程序代表他们调用 销售合作伙伴 API。

亚马逊将混合应用程序视为单个应用程序。例如，当您将混合应用程序发布到商城应用程序商店时，您可以将其作为单个应用程序进行管理。

创建混合的销售合作伙伴 API 应用程序

对于销售合作伙伴 API Beta 计划，您需要亚马逊支持工程师的帮助，以创建混合销售合作伙伴 API 应用程序。使用现有的 Beta 计划支持实例或通过打开新的[支持实例](#)，您可以联系亚马逊支持工程师。请在支持实例的主题行中包含“Selling Partner API Beta”。

创建混合的销售合作伙伴 API 应用程序

1. 将您的亚马逊 MWS 应用程序发布到商城应用程序商店。有关发布您的应用程序的信息，请参阅亚马逊 MWS 文档中的[商城应用程序商店发布指南](#)。
2. 将您的亚马逊 MWS 应用程序注册为销售合作伙伴 API 应用程序。请参阅[注册您的销售合作伙伴 API 应用程序](#)。当您提供注册信息时，请求亚马逊支持工程师将您的亚马逊 MWS 应用程序转换为混合销售合作伙伴 API 应用程序。

您的亚马逊支持工程师将与您联系，确认您的应用程序已注册并转换为草稿状态的混合应用程序。此时，您可以设置和测试 OAuth 授权工作流程。有关更多信息，请参阅[授权销售合作伙伴 API 应用程序](#)。

全球应用程序

您只需在您选择的地区和商城以开发者的身份注册一次，即可创建一个纯粹的 销售合作伙伴 API 应用程序，该应用程序可由任何地区或商城的卖家授权。只要 销售合作伙伴 API 端点与授权您的应用程序的卖家来自同一地区，您只需一组开发者凭据（您的 AWS 访问密钥编码和 AWS 访问密钥）即可调用任何端点。有关如何获取 AWS 访问密钥编码和 AWS 访问密钥的信息，请参阅[第 3 步。创建 IAM 用户](#)。

如果您有混合 销售合作伙伴 API 应用程序，则您对亚马逊商城网络服务（亚马逊 MWS）端点的调用与亚马逊 MWS 应用程序具有相同的限制。也就是说，当您调用亚马逊 MWS 端点时，您必须使用与该端点来自的区域关联的亚马逊 MWS 访问密钥。

有关更多信息，请参阅：

- [纯粹和混合的销售合作伙伴 API 应用程序](#)
- [销售合作伙伴 API 端点](#)

注册您的 销售合作伙伴 API 应用程序

注册您的 销售合作伙伴 API 应用程序，从[第 1 步。创建 AWS 账户](#)开始。注册 销售合作伙伴 API 应用程序后，亚马逊支持工程师将为您提供一个或多个 OAuth 授权 URI。您可以使用 OAuth 授权 URI 来设置 OAuth 授权工作流程。有关更多信息，请参阅[授权销售合作伙伴 API 应用程序](#)。

第 1 步。创建 AWS 账户

您需要 AWS 账户，因为 销售合作伙伴 API 安全模型使用 AWS 身份验证凭据。如果您还不是 AWS 客户，则可以创建一个免费的 AWS 账户。有关更多信息，请参阅[AWS 免费套餐](#)。

第 2 步。创建 AWS IAM 策略

您需要身份和访问管理 (IAM) 策略来定义您将在[第 3 步。创建 IAM 用户](#)中创建的 IAM 用户的权限。IAM 策略将授予 IAM 用户调用 销售合作伙伴 API 的权限。

创建 IAM 策略

1. 登录 AWS 管理控制台并通过 console.aws.amazon.com/iam 打开 AWS 控制台。
2. 在左侧的导航窗格中，单击策略。

如果这是您第一次选择策略，则会显示欢迎使用托管策略页面。单击开始。

3. 单击创建策略按钮。
4. 单击 JSON 选项卡。
5. 将以下代码粘贴到文本框中，替换现有代码，然后单击查看策略。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "execute-api:Invoke",
      "Resource": "arn:aws:execute-api:*:*:*"
    }
  ]
}
```

6. 在查看策略页上，键入要创建的策略的名称和描述（可选）。我们建议您将 IAM 策略命名为 SellingPartnerAPI。
7. 查看策略摘要以了解策略授予的权限，然后单击创建策略按钮。

您的新 IAM 策略将显示在列表中。

有关更多信息，请参阅 AWS 文档中的[创建 IAM 策略](#)。

第 3 步。创建 IAM 用户

您需要 IAM 用户角色来获取 AWS 访问密钥，以便验证对 销售合作伙伴 API 的调用。我们建议专门为此创建一个新的 IAM 用户。

我们会在注册期间将您的 IAM 用户与您的 销售合作伙伴 API 应用程序关联起来。

创建 IAM 用户

1. 登录 AWS 管理控制台并通过 console.aws.amazon.com/iam 打开 AWS 控制台。
2. 在左侧的导航窗格中，单击用户，然后单击添加用户按钮。
3. 键入新用户的用户名。这是 AWS 的登录名。
4. 选择编程访问，然后单击下一步：权限按钮。
5. 在设置权限页上，单击直接附加现有策略按钮。
6. 从策略列表中，选择您在[第 2 步。创建 AWS IAM 策略](#)中创建的策略。



Tip: 对管理的客户进行筛选，使您的策略更容易查找。

7. 单击下一步：查看按钮，查看您截至目前做出的所有选择。当您准备好继续时，请单击创建用户按钮。

显示新 IAM 用户的 AWS 访问密钥编码。

8. 单击显示可查看 AWS 访问密钥。要保存 AWS 访问密钥，请单击下载 .csv，然后将文件保存到安全位置。



Important: 这是您查看或下载 AWS 访问密钥的唯一机会，您将需要使用该密钥验证对销售合作伙伴 API 的调用。将 AWS 访问密钥编码和 AWS 访问密钥保存在安全可靠的位置。在此步骤之后，您将再次无法访问 AWS 访问密钥。如果您丢失了 AWS 访问密钥，则需要创建一个新的 IAM 用户，为其使用自己的一组新密钥。

9. 单击关闭。在用户名列表中，单击您的新 IAM 用户并记录用户 ARN。您将在[第 4 步。提供您的应用程序注册信息](#)时需要它。

有关更多信息，请参阅 AWS 文档中的[在您的 AWS 账户中创建 IAM 用户](#)。

第 4 步。提供您的应用程序注册信息

您需要亚马逊支持工程师的帮助才能注册销售合作伙伴 API 应用程序。您可以通过打开一个[支持实例](#)来就测试版计划接洽亚马逊支持工程师。请在支持实例的主题行中包含“Selling Partner API Beta”。

请使用您的支持实例向亚马逊提供以下信息。

- 您的应用程序的名称。如果您想从您在商城应用程序商店上发布的亚马逊商城网络服务（亚马逊 MWS）应用程序创建混合应用程序，请提供您的亚马逊 MWS 应用程序的名称。有关更多信息，请参阅[混合销售合作伙伴 API 应用程序](#)。
- 您希望卖家登录卖家平台以授权您的应用程序的区域。您将收到您在此处提供的每个区域的特定区域 OAuth 授权 URI。有关更多信息，请参阅[第 0 步。设置 OAuth 授权 URI](#)。
- 您的 IAM 用户的 ARN。请参阅[第 3 步。创建 IAM 用户](#)。
- 登录 URI。这会将用户代理重定向到您网站的登录页面。登录 URI 必须符合 OAuth 2.0 框架。有关更多信息，请参阅 OAuth 2.0 授权框架中的[授权端点](#)。
- 重定向 URI。这将用户代理重定向回您的应用程序。重定向 URI 必须符合 OAuth 2.0 框架。有关详细信息，请参阅 OAuth 2.0 授权框架中的[重定向端点](#)。
- 您的卖家编号。我们需要一个卖家编号来与您的销售合作伙伴 API 应用程序关联，以便您可以登录开发者平台以获取应用程序的开发者信息。这包括您的 IAM ARN 和 LWA 凭据。如果这是针对混合销售合作伙伴 API 应用程序，请提供与您在商城应用程序商店中注册商品应用程序时使用的开发商编号关联的卖家编号。有关更多信息，请参阅[查看您的开发者信息](#)。
- 销售合作伙伴 API 部分。指明您的应用程序将使用的销售合作伙伴 API 部分。
- 亚马逊商城网络服务（亚马逊 MWS）API 部分。指明您的应用程序使用的亚马逊 MWS API 部分。

在我们注册您的应用程序后，亚马逊支持工程师将使用您的支持实例向您提供以下信息：

- 确认我们已注册您的应用程序。

- 您注册的应用程序的应用程序 ID。使用此 ID 可在开发者平台中标识您希望获取 LWA 凭据和 IAM ARN 的应用程序。有关更多信息，请参阅[查看您的开发者信息](#)。
- 适用于您的应用程序的区域 OAuth 授权 URI。当您的应用程序准备好进行测试时，您可以向愿意帮助您测试授权工作流程的卖家提供这些 OAuth 授权 URI 的链接。我们建议您限制对最少数量的卖家进行测试，因为 OAuth 授权 URI 将在您的应用程序发布时发生更改。您也可以使用 OAuth 授权 URI 从您自己的网站启动授权。有关更多信息，请参阅[网站工作流程](#)。

您最多可以接收三个区域 OAuth 授权 URI，每个 URI 对应卖家授权您的应用程序的每个区域。有关区域 URI 的更多信息，请参阅[第 0 步。设置 OAuth 授权 URI](#)。

收到此信息后，您可以查看开发者信息并设置 OAuth 授权工作流程。有关更多信息，请参阅[Viewing your developer information](#)和[Authorizing your 销售合作伙伴 API application](#)。

查看您的开发者信息

[注册销售合作伙伴 API 应用程序](#)后，您可以登录开发者平台以查看您的开发者信息。

查看您的开发者信息

1. 转到与您在[第 4 步。提供您的应用程序注册信息](#)中提供的卖家编号相关联的区域的开发者平台：

- 北美洲：<https://sellercentral.amazon.com/sellingpartner/developerconsole>
- 欧洲：<https://sellercentral.amazon.co.uk/sellingpartner/developerconsole>
- 远东：<https://sellercentral.amazon.co.jp/sellingpartner/developerconsole>

2. 使用上一步中卖家编号的凭据登录。

此时将显示开发者平台页面，其中显示与您的应用程序关联的 IAM ARN

3. 单击 **LWA** 凭据可查看 LWA 客户端标识符和客户端密钥。您将需要这些凭据来请求 LWA 访问令牌。有关更多信息，请参阅[第 1 步。请求使用亚马逊账号登录访问令牌](#)。

授权销售合作伙伴 API 应用程序

销售合作伙伴 API 的授权模式基于 [Login with Amazon](#)，这是亚马逊的 OAuth 2.0 实施。在此模式中，卖家通过与亚马逊和您的网站显示的页面互动来授权您的应用程序。卖家执行的操作会触发您的网站或亚马逊的响应。在每次用户操作时，卖家的浏览器会作为在您的网站和亚马逊之间传递参数的用户代理。要实施 OAuth 授权，您必须将您的网站配置为：(1) 接受并处理亚马逊传递给它的参数，并且 (2) 重定向卖家的浏览器并将参数传递给亚马逊。

卖家可以使用下列其中一个工作流程授权您的应用程序：

- [商城应用程序商店工作流程](#)。从商城应用商店详情页面启动的 OAuth 授权工作流程。
- [网站工作流程](#)。从您自己的网站启动的 OAuth 授权工作流程。

此外，如果卖家授权您代表他们调用亚马逊商城网络服务，您可以使用授权 API 将该授权迁移到混合销售合作伙伴 API 应用程序。这就无需再次请求卖家授权。有关更多信息，请参阅《授权 API 用户指南》。

商城应用程序商店工作流程

商城应用程序商店工作流程是从商城应用程序商店详情页面启动的 OAuth 授权工作流程。当您在商城应用程序商店中发布销售合作伙伴 API 应用程序时，卖家可以通过单击详情页面上的立即授权按钮来授权您的应用程序。

本主题包括商城应用程序商店工作流程步骤以及有关测试工作流程的信息。

测试商城应用程序商店工作流程

在创建生产商城应用程序商店工作流程之前，必须创建可以在草稿状态下授权应用程序的测试工作流程。您的测试工作流程与最终生产工作流程不完全相同。您仍然可以进行测试，以确保您的应用程序可以与亚马逊交换参数并接收授权信息。

以下是测试工作流程与生产工作流程的不同之处：

- 测试工作流程不是从商城应用程序商店详情页面开始，而是从卖家直接导航到应用程序的 OAuth 授权 URI 开始。您可以与合作的可信卖家配合进行测试，也可以使用您自己的销售账户凭据自行测试工作流程。OAuth 授权 URI 必须包含 `version=beta` 参数，指示工作流程授权处于草稿状态的应用程序。当卖家导航到 OAuth 授权 URI 时，工作流程将继续[第 2 步。卖家同意授权您的应用程序](#)。



Note: 如果您有多个区域 OAuth 授权 URI，请务必为卖家提供与他们的销售区域相对应的 OAuth 授权 URI。

- 您的应用程序将 `version=beta` 参数添加到[第 3 步中的亚马逊回调 URI。卖家登录您的网站](#)。这将生成可授权处于“草稿”状态的应用程序的工作流程。

完成工作流程测试后，请更新它使其不再将 `version=beta` 参数添加到[第 3 步中的 Amazon 回调 URI。卖家登录您的网站](#)。这使其成为一个生产工作流程。现在，任何卖家都可以从商城应用程序商店的详情页面开始授权您已发布的应用程序。

生产工作流程从[第 1 步。卖家从商城应用程序商店启动授权](#)开始。

第 1 步。卖家从商城应用程序商店启动授权


1. 卖家登录卖家平台并进入商城应用程序商店。
2. 卖家转到应用程序的详情页面，然后单击立即授权按钮。

此时将显示您的应用程序的同意页面。

第 2 步。卖家同意授权您的应用程序

1. 卖家查看同意页面，查看并接受您的应用程序所请求的数据访问，然后单击立即登录 [您的应用程序名称] 按钮继续。卖家可以单击取消按钮退出，而无需授权。
2. 亚马逊将您的登录 URI（您在应用程序注册时提供）加载到浏览器中，并添加以下查询参数：

参数	描述
amazon_callback_uri	用于将浏览器重定向到亚马逊的 URI。
amazon_state	亚马逊生成的状态值，用于防范跨站点请求伪造攻击。
selling_partner_id	授权您的应用程序的卖家的卖家编号。

 **Note:** 如果这是测试工作流程（卖家通过导航到您的 OAuth 授权 URI 开始），亚马逊将包含 version=beta 参数。如果这是一个生产工作流程（卖家从商城应用程序商店开始），亚马逊不包含该参数。

例如：


```
https://d2zyfnnpjylxu.cloudfront.net/index.html?
amazon_callback_uri=https://amazon.com/apps/authorize/
confirm/amzn1.sellerapps.app.2eca283f-9f5a-4d13-
b16c-474EXAMPLE57&amazon_state=amazonstateexample&selling_partner_id=A3FHEXAMPLEYWS
```

将显示您网站的登录页面。

第 3 步。卖家登录您的网站

1. 卖家登录您的网站。如果卖家还没有账户，他们要完成注册流程。
2. 您的应用程序将 Amazon 回调 URI（在上一步中由亚马逊传递）加载到浏览器中，并添加以下参数：

参数	描述
redirect_uri	用于将浏览器重定向到您的应用程序的 URI。
amazon_state	亚马逊在上一步中传递的 Amazon_State 值。
state	<p>您的应用程序生成的状态值。您的应用程序使用此值来维护此请求和响应之间的状态，从而帮助防范跨站点请求伪造攻击。</p> <p> Important: 由于 OAuth 信息是通过 URI 查询参数传递的，我们强烈建议您执行以下操作：(1) 确保状态令牌对您的用户属于短期令牌，并且具有可验证的唯一性，以及 (2) 设置 Referrer-Policy: no-referrer HTTP 标头，防止将敏感信息泄露到您的网站链接到的网站。有关跨站点请求伪造和计算 state 参数的更多信息，请参阅 Login with Amazon 文档中的跨站点请求伪造。</p>

 **Note:** 如果包含 Version=beta 参数，则工作流程将授权处于草稿状态的应用程序。如果您不包含该参数，则工作流程将授权在商城应用商店中发布的应用程序。

例如：

```
https://amazon.com/apps/authorize/confirm/
amzn1.sellerapps.app.2eca283f-9f5a-4d13-b16c-474EXAMPLE57?
```

```
redirect_uri=https://d2zyfnnpjylxu.cloudfront.net/
landing.html&amazon_state=amazonstateexample&state=-37131022&version=beta
```

或者

```
https://amazon.com/apps/authorize/confirm/
amzn1.sellerapps.app.2eca283f-9f5a-4d13-b16c-474EXAMPLE57?
redirect_uri=https://d2zyfnnpjylxu.cloudfront.net/
landing.html&amazon_state=amazonstateexample&state=-37131022
```

第 4 步。亚马逊向您发送授权信息

卖家平台会短时间显示一个页面，表明亚马逊正在授权您访问卖家数据。显示该页面时，将执行以下操作：

1. 亚马逊将您的重定向 URI 加载到浏览器中，并添加以下查询参数：

参数	描述
state	您在上一步中传递的状态值。
selling_partner_id	授权您的应用程序的卖家的卖家编号。
mws_auth_token	您在为调用亚马逊商城网络服务创建查询字符串时使用的 MWSAuthToken 值。只有当卖家授权混合销售合作伙伴 API 应用程序时，才会传递 mws_auth_token 参数。有关更多信息，请参阅 混合销售合作伙伴 API 应用程序 。
spapi_oauth_code	您为 LWA 刷新令牌交换的 Login with Amazon (LWA) 授权码。有关更多信息，请参阅 第 5 步。您的应用程序用 LWA 授权码交换 LWA 刷新令牌 。

例如：

```
https://client-example.com?state=state-
example&mws_auth_token=mwsauthtokenexample&selling_partner_id=sellingpartneridexample&
spapi_oauth_code=spapioauthcodeexample
```

2. 您的应用程序可以验证 state 值。
3. 您的应用程序会保存 selling_partner_id、mws_auth_token（如果传递）和 spapi_oauth_code 值。
4. 显示网站的登录页面。

第 5 步。您的应用程序用 LWA 授权码交换 LWA 刷新令牌

适用于 JavaScript 的 Login with Amazon SDK 可以帮助您将 LWA 授权码 交换为 LWA 刷新令牌。有关详细信息，请参阅 [Login with Amazon 文档](#)。

- [添加适用于 JavaScript 的使用亚马逊账号登录 SDK](#)
- [授权码授权](#)

将 LWA 授权码 交换为 LWA 刷新令牌

1. 您的应用程序调用 Login with Amazon (LWA) 授权服务器 (<https://api.amazon.com/auth/o2/token>)，将 LWA 授权码 交换为 LWA 刷新令牌。调用必须包含以下查询参数。

参数	描述
grant_type	请求的访问授权类型。必须是 <i>authorization_code</i> 。
code	您在 第 4 步。亚马逊向您发送授权信息 中收到的 LWA 授权码。
redirect_uri	您的应用程序的重定向 URI。

参数	描述
client_id	您的 LWA 凭证的一部分。要获取该值，请参阅 查看您的开发者信息 。
client_secret	您的 LWA 凭证的一部分。要获取该值，请参阅 查看您的开发者信息 。

例如：

```
POST /auth/o2/token HTTP/1.1
Host: api.amazon.com
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
grant_type=authorization_code&code=Sp1x1OexamplebYS6WxSbIA&client_id=foodev&client_secret=foodev
```

2. LWA 授权服务器返回 LWA 刷新令牌。响应采用 JSON 格式并包含以下元素。

参数	描述
access_token	授权您的应用程序代表卖家采取某些操作的令牌。请参阅 连接到 销售合作伙伴 API 。
token_type	返回的令牌类型。应该是 bearer。
expires_in	访问令牌失效之前的秒数。
refresh_token	可以交换为新访问令牌的长期令牌。请参阅 连接到 销售合作伙伴 API 。

```
HTTP/1.1 200 OK
Content-Type: application/json;charset UTF-8
Cache-Control: no-store
Pragma: no-cache
{
  "access_token": "Atza|IQEBLjAsAexampleHpi0U-Dme37rR6CuUpSR",
  "token_type": "bearer",
  "expires_in": 3600,
  "refresh_token": "Atzr|IQEBLzAtAexamplewVz2Nn6f2y-tpJX2DeX"
}
```

3. 您的应用程序保存 refresh_token 值。
4. 浏览器向卖家显示一个页面，指示使用应用程序的后续步骤。

LWA 刷新令牌是您交换 LWA 访问令牌的长期令牌，它必须包含在 销售合作伙伴 API 的每个请求中。发出访问令牌后，它的有效期为一小时。相同的访问令牌可用于多个 API 调用，直到它过期。请参阅[连接到 销售合作伙伴 API](#)。

您的应用程序现已有权代表卖家调用 销售合作伙伴 API。

适用于混合销售合作伙伴 API 应用程序

如果 MWS 身份验证令牌已在 [第 4 步。亚马逊向您发送授权信息](#) 中返回，您的应用程序也有权代表卖家调用 亚马逊商城网络服务。有关更多信息，请参阅[混合 销售合作伙伴 API 应用程序](#)。

网站工作流程

网站工作流是从您自己的网站启动的 OAuth 授权工作流程。卖家登录您的网站，然后单击您配置的用以启动授权的“授权”按钮。有关更多信息，请参阅[第 0 步。设置 OAuth 授权 URI](#)。

本主题包括“网站”工作流程步骤以及有关测试工作流程的信息。

测试网站工作流程

在创建生产网站工作流程之前，必须创建可以在草稿状态下授权应用程序的测试工作流程。这样，您就可以进行测试，以确保您的应用程序可以与亚马逊交换参数并接收授权信息。

以下是测试工作流程与生产工作流程的不同之处：

- 您的应用程序将 `version=beta` 参数添加到[第 1 步。卖家从您的网站启动授权](#)中的 OAuth 授权 URI。这将生成可授权处于“草稿”状态的应用程序的工作流程。

完成工作流程测试后，请更新它使其不再将 `version=beta` 参数添加到[第 1 步。卖家从您的网站启动授权](#)中的 OAuth 授权 URI。这使其成为一个生产工作流程。现在，任何卖家都可以从自己的网站开始授权您已发布的应用程序。

生产工作流程从[第 0 步。设置 OAuth 授权 URI](#)开始。

第 0 步。设置 OAuth 授权 URI

在应用程序网站上设置一个“授权”按钮（或类似的内容），卖家可以单击该按钮启动应用程序的授权。当卖家单击该按钮时，您的网站将 OAuth 授权 URI 加载到浏览器中，卖家将被重定向到卖家平台登录页面。有关获取 OAuth 授权 URI 的信息，请参阅[第 4 步。提供您的应用程序注册信息](#)。


多个区域 OAuth 授权 URI


如果您拥有多个区域的 OAuth 授权 URI，请务必设置“授权”按钮，以便卖家重定向到其所在区域的卖家平台登录页面。有关获取多个区域 URI 的更多信息，请参阅[第 4 步。提供您的应用程序注册信息](#)。

设置“授权”按钮是一次性任务。

第 1 步。卖家从您的网站启动授权

1. 卖家登录您的网站。如果卖家还没有账户，他们要完成注册流程。
2. 卖家单击您在[第 0 步。设置 OAuth 授权 URI](#)中设置的“授权”按钮。如果您有多个区域“授权”按钮，请确保卖家被定向到与其所在销售地区相对应的按钮。
3. 您的应用程序将 OAuth 授权 URI 加载到浏览器中，并添加以下查询参数：

参数	描述
state	<p>您的应用程序生成的状态值。您的应用程序使用此值来维护此请求和响应之间的状态，从而帮助防范跨站点请求伪造攻击。</p> <p> Important: 由于 OAuth 信息是通过 URL 查询参数传递的，我们强烈建议您执行以下操作：(1) 确保状态令牌对您的用户属于短期令牌，并且具有可验证的唯一性，以及 (2) 设置 <code>Referrer-Policy: no-referrer</code> HTTP 标头，防止将敏感信息泄露到您的网站链接到的网站。有关跨站点请求伪造和计算 state 参数的更多信息，请参阅 Login with Amazon 文档中的跨站点请求伪造。</p>

 **Note:** 如果包含 `Version=beta` 参数，则工作流程将授权处于草稿状态的应用程序。如果您不包含该参数，则工作流程将授权在商城应用商店中发布的应用程序。

例如：

```
https://sellercentral.amazon.com/apps/authorize/consent?
application_id=appidexample&state=stateexample&version=beta
```

或者

```
https://sellercentral.amazon.com/apps/authorize/consent?
application_id=appidexample&state=stateexample
```

卖家到达卖家平台的登录页面。

第 2 步。卖家同意授权您的应用程序

1. 卖家登录卖家平台。

此时将显示同意页面。

2. 卖家查看同意页面，查看您的应用程序所请求的数据访问权限，然后单击确认按钮继续。卖家可以单击取消按钮退出，而无需授权。

第 3 步。亚马逊向您发送授权信息

卖家平台会短时间显示一个页面，表明亚马逊正在授权您访问卖家数据。显示该页面时，将执行以下操作：

1. 亚马逊将您的重定向 URI 加载到浏览器中，并添加以下查询参数：

参数	描述
state	第 1 步中的状态值 。卖家从您的网站启动授权。
selling_partner_id	授权您的应用程序的卖家的卖家编号。
mws_auth_token	您在为调用亚马逊商城网络服务创建查询字符串时使用的 MWSAuthToken 值。只有当卖家授权混合销售合作伙伴 API 应用程序时，才会传递 mws_auth_token 参数。有关更多信息，请参阅 混合销售合作伙伴 API 应用程序 。
spapi_oauth_code	您为 LWA 刷新令牌交换的 Login with Amazon (LWA) 授权码。有关更多信息，请参阅 第 4 步。您的应用程序用 LWA 授权码交换 LWA 刷新令牌 。

例如：

```
https://client-example.com?state=state-example&mws_auth_token=mwsauthtokenexample&selling_partner_id=sellingpartneridexample&spapi_oauth_code=spapioauthcodeexample
```

2. 您的应用程序可以验证 state 值。
3. 您的应用程序会保存 selling_partner_id、mws_auth_token（如果传递）和 spapi_oauth_code 值。
4. 显示网站的登录页面。

第 4 步。您的应用程序用 LWA 授权码交换 LWA 刷新令牌

适用于 JavaScript 的 Login with Amazon SDK 可以帮助您将 LWA 授权码交换为 LWA 刷新令牌。有关详细信息，请参阅 Login with Amazon 文档。

- [添加适用于 JavaScript 的使用亚马逊账号登录 SDK](#)
- [授权码授权](#)

将 LWA 授权码 交换为 LWA 刷新令牌

1. 您的应用程序调用 Login with Amazon (LWA) 授权服务器 (<https://api.amazon.com/auth/o2/token>)，将 LWA 授权码 交换为 LWA 刷新令牌。调用必须包含以下查询参数。

参数	描述
grant_type	请求的访问授权类型。必须是 <i>authorization_code</i> 。
code	您在 第 3 步。亚马逊向您发送授权信息 中收到的 LWA 授权码。
redirect_uri	您的应用程序的重定向 URI。
client_id	您的 LWA 凭证的一部分。要获取该值，请参阅 查看您的开发者信息 。
client_secret	您的 LWA 凭证的一部分。要获取该值，请参阅 查看您的开发者信息 。

例如：

```
POST /auth/o2/token HTTP/1.1
Host: api.amazon.com
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
grant_type=authorization_code&code=Sp1x10examplebYS6WxSbIA&client_id=foodev&client_se
```

2. LWA 授权服务器返回 LWA 刷新令牌。响应采用 JSON 格式并包含以下元素。

参数	描述
access_token	授权您的应用程序代表卖家采取某些操作的令牌。请参阅 连接到 销售合作伙伴 API 。
token_type	返回的令牌类型。应该是 bearer。
expires_in	访问令牌失效之前的秒数。
refresh_token	可以交换为新访问令牌的长期令牌。请参阅 连接到 销售合作伙伴 API 。

```
HTTP/1.1 200 OK
Content-Type: application/json;charset UTF-8
Cache-Control: no-store
Pragma: no-cache
{
  "access_token": "Atza|IQEBLjAsAexampleHpi0U-Dme37rR6CuUpSR",
  "token_type": "bearer",
  "expires_in": 3600,
  "refresh_token": "Atzr|IQEBLzAtAexamplewVz2Nn6f2y-tpJX2DeX"
}
```

- 3. 您的应用程序保存 refresh_token 值。
- 4. 浏览器向卖家显示一个页面，指示使用应用程序的后续步骤。

LWA 刷新令牌是您交换 LWA 访问令牌的长期令牌，它必须包含在 销售合作伙伴 API 的每个请求中。发出访问令牌后，它的有效期为一小时。相同的访问令牌可用于多个 API 调用，直到它过期。请参阅[连接到 销售合作伙伴 API](#)。

您的应用程序现已有权代表卖家调用 销售合作伙伴 API。

适用于混合销售合作伙伴 API 应用程序

如果 MWS 身份验证令牌已在 [第 3 步。亚马逊向您发送授权信息](#) 中返回，您的应用程序也有权代表卖家调用 亚马逊商城网络服务。有关更多信息，请参阅[混合 销售合作伙伴 API 应用程序](#)。

自行授权

您可以在开发者平台中自行授权您的应用程序。在执行此操作之前，您必须按照[注册您的销售合作伙伴 API 应用程序](#)中的步骤操作。

要实施完整的 OAuth 授权工作流程，请参阅[授权销售合作伙伴 API 应用程序](#)。

自行授权您的应用程序

1. 转到与您在[第 4 步。提供您的应用程序注册信息](#)中提供的卖家编号相关联的区域的开发者平台：

- 北美：<https://sellercentral.amazon.com/sellingpartner/developerconsole>
- 欧洲：<https://sellercentral.amazon.co.uk/sellingpartner/developerconsole>
- 远东：<https://sellercentral.amazon.co.jp/sellingpartner/developerconsole>

2. 使用上一步中卖家编号的凭据登录。

此时将显示开发者平台页面。

3. 单击要授权的应用程序旁边的编辑 > 授权。

将显示一个页面，其中包含生成刷新令牌按钮。

4. 单击生成刷新令牌。

将显示使用亚马逊账号登录 (LWA) 刷新令牌。您的应用程序现已有权访问您的销售账户。



Important: 单击生成刷新令牌一次以获取刷新令牌，然后将其保存以调用销售合作伙伴 API。如果您多次单击生成刷新令牌，则每次都会获得一个新的刷新令牌，从而使之前的刷新令牌失效。

刷新令牌是您交换短期访问令牌的长期令牌。对销售合作伙伴 API 的每个请求都必须包含访问令牌。发出访问令牌后，它的有效期为一小时。相同的访问令牌可用于多个 API 调用，直到它过期。有关更多信息，请参阅[第 1 步。请求使用亚马逊账号登录访问令牌](#)。

通过 LWA 令牌交换和身份验证生成 Java SDK

这些说明介绍了如何在运行 Microsoft Windows 的计算机上使用 [Swagger 代码生成器](#) 为卖家 API 生成 Java SDK。对于其他操作系统（如 macOS 或 Linux）的用户，该过程是相同的，只是替换掉 Windows 特定的语义（例如 C:\）。虽然这些说明是针对卖家 API 的，但您可以修改说明，以便在销售合作伙伴 API 中为其他 API 创建 SDK。有关每个销售合作伙伴 API 部分的 Swagger 模型，请参阅销售合作伙伴 API GitHub 代码库。

使用此 SDK，您可以使用已为您设置的以下代码调用卖家 API：Login with Amazon 令牌交换（将刷新令牌交换为访问令牌）和身份验证。

使用 LWA 令牌交换和身份验证生成 Java SDK

1. 安装 [Java 8 或更新版本](#)、[Apache Maven 3.6 或更高版本](#) 和 [GNU Wget](#)，并使它们在您的 \$PATH 中可用。
2. 登录到 github.com/amzn。
3. 搜索“amazon-marketplace-api-sdk”，然后单击显示的 [amazon-marketplace-api-sdk](#) 链接。
4. 下载 **Sellers API/Sellers.json** 文件。
5. 下载 **clients/sellingpartner-api-aa-java** 文件夹。此文件夹包含授权和身份验证库，以及用于 Swagger 代码生成器的自定义模板。
6. 打开命令提示符窗口，然后转到要下载 Swagger 代码生成器的目录。
7. 下载最新版本的 Swagger 代码生成器。

例如：

```
wget https://repol.maven.org/maven2/io/swagger/swagger-codegen-cli/2.4.13/swagger-codegen-cli-2.4.13.jar -O swagger-codegen-cli.jar
```

swagger-codegen-cli.jar 下载到当前目录。



Note: 您也可以使用浏览器访问此网址来从 maven.org 下载：<https://repol.maven.org/maven2/io/swagger/swagger-codegen-cli/2.4.13/swagger-codegen-cli-2.4.13.jar>

8. 将 **Sellers.json** 和 **swagger-codegen-cli.jar** 复制到一个对您有意义的目录结构中。在此示例中，我们将它们复制到 C:\SwaggerToCL。
9. 根据 **clients/sellingpartner-api-aa-java** 文件夹中的模板生成 SDK。

例如：

```
java -jar C:\SwaggerToCL\swagger-codegen-cli.jar generate -i C:\SwaggerToCL\Sellers.json -l java -t [path to clients\sellingpartner-api-aa-java directory]/resources/swagger-codegen/templates/ -o C:\SwaggerToCL\Sellers_JavaCL
```

将 SDK 复制到 C:\SwaggerToCL\Sellers_JavaCL

10. 构建 AA 库并将其添加为 SDK 的依赖项：

- a. 导航到 **clients/sellingpartner-api-aa-java** 文件夹并运行 mvn 包。这会生成一个名为“target”的文件夹。在这个文件夹中，有一个名为 **sellingpartnerapi-aa-java-1.0-jar-with-dependencies.jar** 的 JAR 文件（或类似文件）和所有必需的依赖项。
- b. 在本地 Maven 存储库中安装 JAR 文件。

例如：

```
mvn install:install-file -Dfile=[path to JAR file in "target" folder] -DgroupId=com.amazon.sellingpartnerapi -DartifactId=sellingpartnerapi-aa-java -Dversion=1.0 -Dpackaging=jar
```

您可以在 **clients/sellingpartner-api-aa-java** 文件夹中的 **pom.xml** 文件顶部附近找到实际的 `groupId`、`artifactId` 和 `version` 值。

- c. 在客户端库的 **pom.xml** 中添加 AA 库的依赖项：

例如：

```
<dependency>
  <groupId>com.amazon.sellingpartnerapi</groupId>
  <artifactId>sellingpartnerapi-aa-java</artifactId>
  <version>1.0</version>
</dependency>
```

生成软件 SDK 后，您可以使用它调用销售合作伙伴 API。请参阅[使用生成的 Java SDK 连接到销售合作伙伴 API](#)。

使用生成的 Java SDK 连接到销售合作伙伴 API

您必须首先注册您的应用程序，并且必须获得卖家授权，之后才能将该应用程序连接到销售合作伙伴 API。请参阅[注册您的销售合作伙伴 API 应用程序](#)和[授权销售合作伙伴 API 应用程序](#)。

这些说明介绍了如何使用生成的 Java SDK 调用销售合作伙伴 API。SDK 公开用于配置 LWA 和 AWS 凭据的类，使用这些类来交换 LWA 令牌并为您签署请求。有关更多信息，请参阅[通过 LWA 令牌交换和身份验证生成 Java SDK](#)。

第 1 步。配置 AWS 凭据

使用以下参数创建 `AWSAuthenticationCredentials` 的实例：

名称	描述	必需项
<code>accessKeyId</code>	您的 AWS 访问密钥 ID。有关更多信息，请参阅 第 3 步。创建 IAM 用户 。	是
<code>secretKey</code>	您的 AWS 访问密钥。有关更多信息，请参阅 第 3 步。创建 IAM 用户 。	是
<code>region</code>	您的调用所指向的 AWS 区域。有关更多信息，请参阅 销售合作伙伴 API 端点 。	是

例如：

```
import com.amazon.SellingPartnerAPIAA.AWSAuthenticationCredentials;

...

AWSAuthenticationCredentials
awsAuthenticationCredentials=AWSAuthenticationCredentials.builder()
    .accessKeyId("myAccessKeyId")
    .secretKey("mySecretId")
    .region("us-east-1")
    .build();
```

第 2 步。配置 LWA 凭据

使用以下参数创建 `LWAAuthorizationCredentials` 的实例：

名称	描述	必需项
<code>clientId</code>	您的 LWA 客户端标识符。有关更多信息，请参阅 查看您的开发者信息 。	是
<code>clientSecret</code>	您的 LWA 客户密钥。有关更多信息，请参阅 查看您的开发者信息 。	是
<code>refreshToken</code>	LWA 刷新令牌。当卖家授权您的应用程序时，可以获得该值。有关更多信息，请参阅 授权销售合作伙伴 API 应用程序 。	是
<code>endpoint</code>	LWA 身份验证服务器 URI。	是

例如：

```
import com.amazon.SellingPartnerAPIAA.LWAAuthorizationCredentials;

...
```

```
LWAAuthorizationCredentials lwaAuthorizationCredentials =  
    LWAAuthorizationCredentials.builder()  
        .clientId("myClientId")  
        .clientSecret("myClientSecret")  
        .refreshToken("Aztr|...")  
        .endpoint("https://api.amazon.com/auth/o2/token")  
        .build();
```

第 3 步。创建卖家 API 实例并调用操作

通过配置 `AWSAuthenticationCredentials` 和 `LWAAuthorizationCredentials` 实例，您可以创建一个 `SellersApi` 实例并调用操作。

例如：

```
SellersApi sellersApi = new SellersApi.Builder()  
    .awsAuthenticationCredentials(awsAuthenticationCredentials)  
    .lwaAuthorizationCredentials(lwaAuthorizationCredentials)  
    .endpoint(https://sellingpartnerapi-na.amazon.com)  
    .build();  
sellersApi.getMarketplaceParticipations();
```

生成 Java 客户端库

这些说明介绍了如何在运行 Microsoft Windows 的计算机上使用 [Swagger 代码生成器](#) 为卖家 API 生成 Java 客户端库。对于其他操作系统（如 macOS 或 Linux）的用户，该过程是相同的，只是替换掉 Windows 特定的语义（例如 C:\）。虽然这些说明是针对卖家 API 的，但您可以修改说明，以便在销售合作伙伴 API 中为其他 API 创建客户端库。有关每个销售合作伙伴 API 部分的 Swagger 模型，请参阅销售合作伙伴 API GitHub 代码库。

虽然生成的客户端库可以帮助您调用销售合作伙伴 API，但它不包含用于 LWA 令牌交换和身份验证的代码。对于这一点，请参阅[第 1 步。请求使用亚马逊账号登录访问令牌](#)和[第 4 步。创建并签署您的请求](#)。或者，有关包含 LWA 令牌交换和身份验证的 SDK，请参阅[通过 LWA 令牌交换和身份验证生成 Java SDK](#)。

生成 Java 客户端库

1. 安装 [Java 8 或更新版本](#)、[Apache Maven 3.6 或更高版本](#)和 [GNU Wget](#)，并使它们在您的 \$PATH 中可用。
2. 登录到 github.com/amzn。
3. 搜索“amazon-marketplace-api-sdk”，然后单击显示的 **amazon-marketplace-api-sdk** 链接。
4. 下载 **Sellers API/Sellers.json** 文件。
5. 打开命令提示符窗口，然后导航到要下载 Swagger 代码生成器的目录。
6. 下载最新版本的 Swagger 代码生成器。

例如：

```
wget https://repol.maven.org/maven2/io/swagger/swagger-codegen-cli/2.4.13/swagger-codegen-cli-2.4.13.jar -O swagger-codegen-cli.jar
```

swagger-codegen-cli.jar 下载到当前目录。



Note: 您也可以使用浏览器访问此网址来从 maven.org 下载：<https://repol.maven.org/maven2/io/swagger/swagger-codegen-cli/2.4.13/swagger-codegen-cli-2.4.13.jar>

7. 将 **Sellers.json** 和 **swagger-codegen-cli.jar** 复制到一个对您有意义的目录结构中。在此示例中，我们将它们复制到 C:\SwaggerToCL。
8. 生成客户端库。

例如：

```
java -jar C:\SwaggerToCL\swagger-codegen-cli.jar generate -i C:\SwaggerToCL\Sellers.json -l java -o C:\SwaggerToCL\Sellers_JavaCL
```

客户端库被复制到 C:\SwaggerToCL\Sellers_JavaCL。

生成客户端库后，您可以使用它来帮助您调用销售合作伙伴 API。请参阅[连接到销售合作伙伴 API](#)。

连接到 销售合作伙伴 API

您必须首先注册您的应用程序，并且必须获得卖家授权，之后才能将该应用程序连接到销售合作伙伴 API。请参阅[注册您的 销售合作伙伴 API 应用程序](#)和[授权销售合作伙伴 API 应用程序](#)。

这些说明向您展示了调用销售合作伙伴 API 的步骤。有关构建销售合作伙伴 API URI 并向其添加标头的帮助，请参阅[生成 Java 客户端库](#)。有关用来交换 LWA 令牌及执行身份验证的代码的更完整解决方案，请参阅[通过 LWA 令牌交换和身份验证生成 Java SDK](#)。

第 1 步。请求使用亚马逊账号登录访问令牌

Login with Amazon (LWA) 访问令牌 授权您的应用程序代表卖家采取某些操作。LWA 访问令牌 在发出后一小时过期，并且必须包含在每个 销售合作伙伴 API 请求中。


要请求 LWA 访问令牌，请使用以下参数向 LWA 身份验证服务器 (<https://api.amazon.com/auth/o2/token>) 发出安全的 HTTP POST：

名称	描述	必需项
grant_type	请求的访问授权类型。必须是 <i>refresh_token</i> 。	是
refresh_token	当卖家授权您的应用程序时，可以获得该值。有关更多信息，请参阅 授权销售合作伙伴 API 应用程序 。	是
client_id	注册应用程序时获取该值。请参阅 看您的开发者信息 。	是
client_secret	注册应用程序时获取该值。请参阅 看您的开发者信息 。	是

例如：

```
POST /auth/o2/token HTTP/1.1
Host: api.amazon.com
Content-Type: application/x-www-form-urlencoded;charset=UTF-8

grant_type=refresh_token
&refresh_token=Atzr|IQEBLzAtAhRPpMJxdwVz2Nn6f2y-tpJX2DeX...
&client_id=foodev
&client_secret=Y76SDl2F
```

 **Tip:** 要避免在调用 LWA 授权服务器时出现不可信的证书颁发机构 (CA) 错误，请务必更新您的信任存储，以便您的应用程序信任 LWA 授权服务器。

响应

成功的响应包括以下值。

名称	描述
access_token	LWA 访问令牌。最大尺寸： 2048 个字节。
token_type	返回的令牌类型。必须是 <i>bearer</i> 。
expires_in	LWA 访问令牌 无效之前的秒数。
refresh_token	您在请求中提交的 LWA 访问令牌。最大尺寸： 2048 个字节。

```
HTTP/1.1 200 OK
```

```
Content-Type: application/json;charset UTF-8
Cache-Control: no-store
Pragma: no-cache
{
  "access_token": "Atza|IQEBLjAsAhRmHjNgHpi0U-Dme37rR6CuUpSREXAMPLE",
  "token_type": "bearer",
  "expires_in": 3600,
  "refresh_token": "Atzr|IQEBLzAtAhRPpMJxdwVz2Nn6f2y-tpJX2DeXEXAMPLE"
}
```

有关更多信息，请访问 [Login with Amazon](#) 文档中的[授权代码授权](#)页面。

第 2 步。构建 销售合作伙伴 API URI

以下是 销售合作伙伴 API URI 的组成部分。

名称	描述	示例
HTTP 方法	其中一个 销售合作伙伴 API HTTP 方法 。	GET
端点	一个 销售合作伙伴 API 端点 。	<code>https://sellingpartnerapi-na.amazon.com</code>
路径	销售合作伙伴 API 部分/版本。部分/资源的数量。	<code>/sellers/v1/marketplaceParticipations</code>
查询字符串	查询参数。	<code>?marketplace=ATVPDKIKX0DER</code>

例如：

```
GET https://https://sellingpartnerapi-na.amazon.com/sellers/v1/
marketplaceParticipations?marketplace=ATVPDKIKX0DER
```

第 3 步。将标头添加到 URI

将标头添加到您在[第 2 步。构建 销售合作伙伴 API URI](#)中构建的 URI。

以下是您在 销售合作伙伴 API 请求中包含的 HTTP 标头：

请求标头

名称	描述
host	商城端点。请参阅 销售合作伙伴 API HTTP 方法 。
x-amz-access-token	LWA 访问令牌。请参阅 第 1 步。请求使用亚马逊账号登录访问令牌 。
x-amz-date	请求的日期和时间。
user-agent	您的应用程序名称和版本号、平台和编程语言。这些内容可帮助亚马逊诊断和修复您可能遇到的服务问题。请参阅 在所有请求中包含 User-Agent 标头参数 。


以下是销售合作伙伴 API 的请求示例，其中包含 URI 和标头，但没有签名信息：

```
GET https://sellingpartnerapi-na.amazon.com/sellers/v1/
marketplaceParticipations?marketplace=ATVPDKIKX0DER HTTP/1.1
host: sellingpartnerapi-na.amazon.com
user-agent: My Selling Tool/2.0 (Language=Java/1.8.0.221;
Platform=Windows/10)
x-amz-access-token=Atza|IQEBLjAsAhRmHjNgHpi0U-Dme37rR6CuUpSREXAMPLE
x-amz-date: 20190430T123600Z
```

要为对于 销售合作伙伴 API 的请求签名，请参阅[第 4 步。创建并签署您的请求](#)。

第 4 步。创建并签署您的请求

销售合作伙伴 API 使用 AWS [签名版本 4 签名流程](#)对请求进行身份验证。当您向 销售合作伙伴 API 发送 HTTP 请求时，您需要对请求进行签名，以便亚马逊能够识别请求发送者。您可以使用 AWS 访问密钥签署请求，该密钥包括访问密钥 ID 和访问密钥。有关获取 AWS 访问密钥的信息，请参阅[第 3 步。创建 IAM 用户](#)。

 **Note:** 您需要了解如何在手动创建 HTTP 请求时对其进行签名。当您使用其中一个 AWS SDK 为您计算签名时，SDK 会自动使用您在配置它时指定的 AWS 访问密钥对请求进行签名。您使用 SDK 时，无需了解如何自己签署请求。例如，Java 开发者可以使用适用于 Java 的 AWS SDK 中的 [AWS4Signer.java](#) 作为计算签名的模型。您可以在 [AWS GitHub 代码库](#) 中找到其他语言的 SDK。

要创建并签署您的请求，请完成以下操作：

1. 创建规范请求

按照 AWS 文档的[任务 1：创建签名版本 4 的规范请求](#)中的说明操作，使用该指南：

- 有关创建规范请求时开始的未签名请求示例，请参阅[第 3 步。将标头添加到 URI](#)。
- 使用 SHA-256 作为哈希算法。
- 请勿将身份验证信息放在查询参数中。将其放在 Authorization 标头参数中。有关使用 Authorization 标头参数作为身份验证信息的信息，请参阅[Authorization 标头](#)。

2. 创建要签名的字符串

按照 AWS 文档的[任务 2：为签名版本 4 创建要签名的字符串](#)中的说明操作，使用该指南：

- 算法指定值为 AWS4-HMAC-SHA256
- 要确定凭据范围，请参阅[Credential scope](#)。

3. 计算签名

按照 AWS 文档的[任务 3：计算 AWS 签名版本 4 的签名](#)中的说明操作。

 **Important:** 请参阅[Credential scope](#)部分以帮助完成此步骤。

4. 添加签名信息

按照 AWS 文档的[任务 4：将签名添加到 HTTP 请求](#)中的说明操作，使用该指南：

- 请勿将签名信息添加到查询字符串。将其添加到 Authorization 标头参数。
- 有关创建 Authorization 标头参数的详细信息，请参阅[Authorization 标头](#)。

以下示例显示了在您使用 Authorization 标头向其添加签名信息后的类似请求。

```
GET /sellers/v1/marketplaceParticipations?marketplace=ATVPDKIKX0DER
HTTP/1.1
Authorization: AWS4-HMAC-SHA256Credential=AKIDEXAMPLE/20190430/us-east1/
execute-api/aws4_request, SignedHeaders=host;user-agent;x-amz-access-token,
Signature=5d672d79c15b13162d9279b0855cfba6789a8edb4c82c400e06b5924aEXAMPLE
https://sellingpartnerapi-na.amazon.com
user-agent: My Selling Tool/2.0 (Language=Java/1.8.0.221;
Platform=Windows/10)
x-amz-access-token=Atza|IQEBLjAsAhRmHjNgHpi0U-Dme37rR6CuUpSREXAMPLE
x-amz-date: 20190430T123600Z
```

凭据范围


凭据范围是您在签署 销售合作伙伴 API 请求时创建的“要签名的字符串”的组成部分。请参阅[创建并签署您的请求](#)。

凭据范围由斜杠分隔的各组成部分的字符串表示，如下表所示：

组成部分	描述	示例
日期	一个八位数的字符串，表示请求的年份 (YYY)、月份 (MM) 和日期 (DD)。	20190430
AWS 区域	您要向其发送请求的地区。请参阅 销售合作伙伴 API endpoints 。	us-east-1
服务	您请求的服务。您可以在端点中找到该值。请参阅 销售合作伙伴 API endpoints 。	execute-api
终止字符串	一个特殊的终止字符串。对于 AWS 签名版本 4，值为 aws4_request	aws4_request

例如：

```
20190430/us-east-1/execute-api/aws4_request
```

 **Important:** 作为凭据范围一部分使用的日期必须与您的请求日期匹配，如 x-amz-date 标头中指定的那样。有关更多信息，请参阅 AWS 文档中的[处理签名版本 4 中的日期](#)。

有关更多信息，请参阅[第 4 步。创建并签署您的请求](#)。

Authorization 标头

Authorization 标头包含请求的签名信息。尽管标头命名为“Authorization”，但签名信息可用于身份验证。

下面列出了 Authorization 标头的组成部分：

组成部分	描述
用于签名的算法	整个签名过程中使用的哈希算法。销售合作伙伴 API 需要 SHA-256。您可以在 第 4 步。创建并签署您的请求 中指定此项。
Credential	您的 AWS 访问密钥编码加上 Credential scope 。您可以在 第 3 步。创建 IAM 用户 中获取 AWS 访问密钥编码。
SignedHeaders	签名请求中包含的所有 HTTP 标头的列表。有关示例，请参阅 第 3 步。将标头添加到 URI 。
Signature	在 第 4 步。创建并签署您的请求 中计算的签名。

例如：

```
Authorization: AWS4-HMAC-SHA256 Credential=AKIDEXAMPLE/20190430/us-east1/execute-api/aws4_request, SignedHeaders=host;user-agent;x-amz-accesstoken;x-amz-date, Signature=5d672d79c15b13162d9279b0855cfba6789a8edb4c82c400e06b5924aEXAMPLE
```

有关更多信息，请参阅[第 4 步。创建并签署您的请求](#)。

响应格式

为响应 HTTP 请求，销售合作伙伴 API 返回响应标头和 JSON 响应消息。

响应标头

名称	描述
Content-Length	标准 HTTP 响应标头。
Content-Type	标准 HTTP 响应标头。
Date	标准 HTTP 响应标头。
x-amzn-RequestId	请求编码。如果您与我们联系以获得支持，请包括此内容。

成功响应

如果您的请求成功，则 销售合作伙伴 API 返回请求的数据。以下是一个成功响应的例子：

```
HTTP/1.1 200 OK
Content-Length: 368
Content-Type: application/json
Date: Thu, 0Jun 2012 22:23:31 GMT
x-amzn-RequestId: 6875f61f-6aa1-11e8-98c6-9b9a3a7283a4
{
  "payload": [
    {
      "marketplace": {
        "id": "ATVPDKIKX0DER",
        "countryCode": "US",
        "name": "Amazon.com",
        "defaultCurrencyCode": "USD",
        "defaultLanguageCode": "en_US",
        "domainName": "www.amazon.com"
      },
      "participation": {
        "isParticipating": true,
        "hasSuspendedListings": false
      }
    }
  ]
}
```

错误响应

如果您的请求不成功，则 销售合作伙伴 API 返回错误响应。以下是错误响应中响应消息的元素：

响应消息

元素	描述	必需项
code	HTTP 状态代码。	是
message	错误条件的说明。	是
details	其他信息的链接。	否

以下是错误响应的示例：

```
HTTP/1.1 400 Bad Request
Content-Length: 117
Content-Type: application/json
Date: Fri, 0Jun 20100:48:2GMT
x-amzn-ErrorType: ValidationException
x-amzn-RequestId: a8c8d99a-6ab5-11e8-b0f8-19363980175b
{
  "errors": [
    {
      "message": "Access to requested resource is denied.",
      "code": "Unauthorized",
      "details": "Access token is missing in the request header."
    }
  ]
}
```

免授权操作

免授权操作是您可以在没有销售合作伙伴明确授权的情况下调用的操作。如果您调用其他销售合作伙伴 API 操作，必须要接收和交换 LWA 授权代码并刷新令牌才能获取 LWA 访问令牌，而使用此授权模式时不需要这样做。相反，您只需调用一次 LWA 授权服务器，即可获得 LWA 访问令牌。

目前授权 API 的 GET/authorization/v1/authorizationCode 操作是销售合作伙伴 API 中唯一的免授权操作。

要调用免授权操作，请按照[连接到销售合作伙伴 API](#)中的说明操作。使用此重要变体：将[第 1 步。请求使用亚马逊账号登录访问令牌](#)中的说明替换为以下部分中的说明：请求使用亚马逊账号登录访问令牌以调用免授权操作。

请求使用亚马逊账号登录访问令牌以调用免授权操作

使用亚马逊账号登录 (LWA) 访问令牌授权您的应用程序代表销售合作伙伴执行某些操作。LWA 访问令牌在发出后一小时过期，并且必须包含在销售合作伙伴 API 的每个请求中。

要请求 LWA 访问令牌，请使用以下参数向 LWA 身份验证服务器 (<https://api.amazon.com/auth/o2/token>) 创建一个安全的 HTTP POST：

名称	描述	必需项
grant_type	请求的访问授权类型。必须是 client_credentials。	是
scope	授权范围。请求 LWA 访问令牌以调用授权 API 的 GET/authorization/v1/authorizationCode 操作时，该值必须为 sellingpartnerapi::migration。	是
client_id	注册应用程序时获取该值。请参阅 看您的开发者信息 。	是
client_secret	注册应用程序时获取该值。请参阅 看您的开发者信息 。	是

例如：

```
POST /auth/o2/token HTTP/1.1
Host: api.amazon.com
Content-Type: application/x-www-form-urlencoded;charset=UTF-8

grant_type=client_credentials
&scope=sellingpartnerapi::migration
&client_id=foodev
&client_secret=Y76SDl2F
```

限制：针对提交请求频率的限制

限制是限制应用程序在给定时间段内对 销售合作伙伴 API 中的操作发出的请求数量的过程。限制功能可以避免网络服务遭受过量请求的不利影响，帮助确保所有授权的应用程序均可访问网络服务。

销售合作伙伴 API 的相关限制适用于每个操作的每个卖家-应用程序对。这意味着限制卖家-应用程序对的一个操作不会影响对该卖家-应用程序对的任何其他操作的限制。

销售合作伙伴 API 使用[令牌存储桶算法](#)限制请求，其中令牌针对请求进行计数。该服务设置了稳态速率的限制，并设置针对每个操作的突增请求提交。在令牌存储桶算法中，突增是最大存储桶大小。

销售合作伙伴 API 的相关限制

操作	速率（每分钟请求数）	突增
GET/authorization/v1/authorizationCode	60	5
GET/sales/v1/orderMetrics	.5	15
GET/sellers/v1/marketplaceParticipations	.016	15

在所有请求中包含 **User-Agent** 标头

User-Agent 标头可标识您的应用程序、其版本号以及您正在使用的平台和编程语言。您必须在提交给的销售合作伙伴 API 每个请求中包含一个 **User-Agent** 标头。这样可帮助亚马逊更有效地判断和解决问题，也有助于您更好地使用该服务。

要创建 **User-Agent** 标头，请以应用程序名称开头，然后依次是正斜杠、应用程序版本、空格、左括号、**Language** 名称值对，以及右括号。**Language** 参数是必需的属性，但您还可以添加其他属性，并以分号隔开。

以下伪代码展示了在最低限度上可接受的 **User-Agent** 标头：

```
AppId/AppVersionId (Language=LanguageNameAndOptionallyVersion)
```

以下是应用程序集成商可能使用的 **User-Agent** 标头示例：

```
My Selling Tool/2.0 (Language=Java/1.8.0.221; Platform=Windows/10)
```

如果您是通过自己的 IT 部门进行集成的大卖家，请考虑创建包含 **Host** 属性的 **User-Agent** 标头，如以下示例所示。这可以帮助亚马逊支持工程师为您更有效地解决问题。

```
MyCompanyName/build1611 (Language=Perl; Host=jane.desktop.example.com)
```


要指定其他属性，请使用以下格式：**AttributeName=Value;**，将每个名称值对用分号隔开。如果您需要使用反斜杠 (\)，请用另一个反斜杠 (\\) 引用它。同样，在应用程序名称中把一个正斜杠括住 (\/)，在应用程序版本中包含一个左括号 (\()，在属性名称中包含一个等号 (\=)，在属性值中包含一个右括号 (\)) 和分号 (\;)。

由于 **User-Agent** 标头要包含在每个请求中进行传送，因此建议您限制标头的长度。如果 **User-Agent** 标头长度超过 500 个字符，则销售合作伙伴 API 会拒绝该标头。

销售合作伙伴 API 沙箱

销售合作伙伴 API 提供了一个沙箱环境，允许您在不影响生产数据或触发真实情况下测试应用程序。对销售合作伙伴 API 进行沙箱调用与进行生产调用相同，不同之处只有您要将调用指向[销售合作伙伴 API 沙箱端点](#)。调用沙箱端点会返回所有销售合作伙伴 API 的静态模拟响应。您可以在 Swagger 模型 JSON 文件中为要调用的 API 引用这些模拟响应。有关更多信息，请参阅[如何对销售合作伙伴 API 进行沙箱调用](#)。

销售合作伙伴 API 沙箱的工作原理与许多模拟框架相同，因为它使用模式匹配来返回指定参数存在时指定的响应。开发者在发送与指定参数匹配的请求时会收到在 x-amazon-spds-sandbox-behaviors 对象中定义的响应。如果 API 需要未在 x-amazon-spds-sandbox-behaviors 对象中指定的任何参数，则只要请求有效，沙箱都会提供响应，而不考虑请求中的参数值。

 **Important:** 沙箱用于测试功能，而非测试可扩展性。对沙箱端点的调用受以下限制的约束：速率 = 每秒 5 个请求；突增 = 15。有关限制的更多信息，请参阅[限制：针对提交请求频率的限制](#)。

对于销售合作伙伴 API 的公开发布版本，我们将包含沙箱调用的输入验证，以便您收到无效请求的真实错误消息。

如何对销售合作伙伴 API 进行沙箱调用

第 1 步。获取您想要调用的 API 的 Swagger 模型 JSON 文件

1. 转到 [Github.com/amzn](https://github.com/amzn)。
2. 搜索 “amazon-marketplace-api-sdk”，然后单击显示的 [amazon-marketplace-api-sdk](#) 链接。
3. 下载要调用的 API 的 JSON 文件。

第 2 步。检查 JSON 文件中的请求参数

1. 打开您在上一步中下载的 JSON 文件。
2. 搜索 “x-amazon-spds-sandbox-behaviors”。

JSON 文件的 x-amazon-spds-sandbox-behaviors 对象包含对 API 的沙箱调用的请求和响应示例。如果请求示例包含参数，请在以下步骤中使用它们。

第 3 步。对 API 进行沙箱调用

按照与执行生产调用时相同的方式对 API 进行沙箱调用，但有以下不同之处：

1. 如果 JSON 文件的 x-amazon-spds-sandbox-behaviors 对象中的请求对象包含一个或多个参数/值对，请在调用中指定这些对象。
2. 将您的调用指向[销售合作伙伴 API 沙箱端点](#)之一。

您应收到一个响应，该响应与 JSON 文件的 x-amazon-spds-sandbox-behaviors 对象中包含的有效负载对象匹配。

销售合作伙伴 API 沙箱端点

销售合作伙伴 API 具有针对北美、欧洲和远东销售区域的沙箱端点。有关更多信息，请参阅[销售合作伙伴 API 沙箱](#)。

销售区域	端点	AWS 区域
北美（加拿大、美国、墨西哥和巴西商城）	https://sandbox.sellingpartnerapi-na.amazon.com	us-east-1
欧洲（西班牙、英国、法国、德国、意大利、土耳其、阿联酋和印度商城）	https://sandbox.sellingpartnerapi-eu.amazon.com	eu-west-1
远东（新加坡、澳大利亚和日本市场）	https://sandbox.sellingpartnerapi-fe.amazon.com	us-west-2

常见问题

销售合作伙伴 API 与亚马逊商城网络服务有何不同？

尽管销售合作伙伴 API 和亚马逊商城网络服务（亚马逊 MWS）都是支持以编程方式访问卖家数据的 Web 服务，但存在着显著差异。以下是销售合作伙伴 API 与亚马逊 MWS 之间的一些主要区别：

- 销售合作伙伴 API 将数据视为符合 REST 的资源，可以通过标准 HTTP 方法访问和修改。相比之下，亚马逊 MWS 使用特定于亚马逊 MWS 的操作公开数据。
- 销售合作伙伴 API 授权利用 LWA，即亚马逊对 OAuth 2.0 的实施。此模型无需按照亚马逊 MWS 的要求手动交换身份验证令牌。请参阅[授权您的销售合作伙伴 API 应用程序](#)。
- 通过亚马逊 MWS，卖家授权开发者。使用销售合作伙伴 API，卖家授权应用程序。使用销售合作伙伴 API，开发者可以创建多个需要不同级别的卖家数据访问权限的应用程序。
- 销售合作伙伴 API 提供比亚马逊 MWS 更精细的粒度数据访问控制。开发者只能请求访问他们所需的数据，卖家可以授予 API 部分、操作或数据资源级别的权限。
- 销售合作伙伴 API 允许您使用 AWS 身份和访问管理 (IAM) 直接获取和管理自己的身份验证凭据。借助亚马逊 MWS，您将使用特殊注册工作流程接收亚马逊提供的身份验证凭据，并通过与亚马逊 MWS 支持部门的帮助打开联系人来获得新凭据。请参阅[第 3 步。创建 IAM 用户](#)。
- 销售合作伙伴 API 使用 AWS 签名版本 4 进行身份验证。亚马逊 MWS 使用签名版本 2。请参阅[第 4 步。创建并签署您的请求](#)。