

Amazon Marketplace

Last Modified on 01/19/2021 1:00 pm EST

amazon.com

Amazon Marketplace

Amazon Marketplace Connector

On this page:

Authentication Prerequisites

The Amazon Marketplace connector uses the **aws-v2** authentication workflow. Before you can authenticate an instance of the Amazon Marketplace connector, you need to have registered with Amazon Marketplace Web Service (Amazon MWS) as a developer and have a seller and developer account. To authenticate with the Amazon Marketplace connector you will need:

- AWS API Key
- AWS API Secret
- MWS Auth Token
- Marketplace Id
- Marketplace Seller ID

To locate or generate the above credentials, refer to [Amazon User Guide](#).

Your **AWS Access Key ID** will be the **AWS API Key** and your **AWS Secret Access Key** will be the **AWS API Secret**.

Required Authentication Parameters

You can authenticate a connector instance using the UI or via API calls. For detailed information on how to authenticate an instance, see our related documentation:

- [Authenticate a Connector Instance \(UI\)](#)
- [Authenticate a Connector Instance \(API\)](#)

To authenticate an instance of the Amazon Marketplace connector, you will need the following parameters:

Parameter Name/UI Value	API Value	Description	Source
AWS API Key	oauth.api.key	Generated before authentication	Amazon User Guide
AWS API Secret	oauth.api.secret	Generated before authentication	Amazon User Guide
MWS Auth Token	mws.auth.token	Generated before authentication	Amazon User Guide
Marketplace id	marketplace.id	Generated before authentication	Amazon User Guide
Marketplace Seller id	marketplace.seller.id	Generated before authentication	Amazon User Guide
OAuth Callback URL	oauth.callback.url	Default value is https://auth.cloudelements.io/oauth	Authenticate a Connector Instance (API)

Sample Configuration JSON

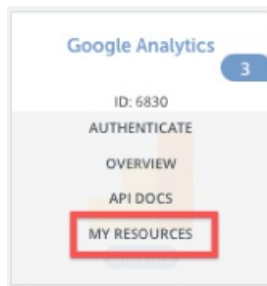
```
"configuration": {  
  "aws.api.key": "[myAWSApiKey]",  
  "aws.api.secret": "[myAWSApiSecret]",  
  "mws.auth.token": "[myAWSauthToken]",  
  "marketplace.id": "[myMarketplaceId]",  
  "marketplace.seller.id": "[myMarketplaceSellerId]",  
  "oauth.callback.url": "https://auth.cloudelements.io/oauth"  
}
```

Connector Fact Sheet and Reference

Authentication and Configuration Parameters

To see all authentication parameters for the Amazon Marketplace connector, follow these steps:

1. Sign in to SAP Cloud Platform Open Connectors and navigate to Connectors.
2. Hover over the card for a connector, and then click My Resources.



3. In the top navigation toolbar, click Setup.
4. From the Setup page, you can view a complete list of the connector's authentication and configuration parameters.

Events and Supported Resources

Amazon Marketplace connector supports events via polling. For detailed information about our Events framework and how to configure a connector instance with events, see our documentation:

- [Authenticate a Connector Instance with Events \(UI\)](#)
- [Authenticate a Connector Instance with Events \(API\)](#)

You can set up events via polling for the `orders` resource of the Amazon Marketplace connector.

Queryable Endpoints

You can use OCNQL to query the following endpoints of the Amazon Marketplace connector:

- GET /amazon-fulfillments
- GET /bulk/jobs
- GET /feeds
- GET /inventories/amazon-fulfillments
- GET /{objectName}
- GET /{objectName}/{objectId}/{childObjectName}
- GET /orders

- GET /products
- GET /reports
- GET /requested-reports
- GET /transaction-groups
- GET /transactions

Connector API Documentation

The base URL for all API requests is

```
https://api.openconnectors.us2.ext.hana.ondemand.com/elements/api-v2 .
```

Provider Documentation

- [Amazon MWS User Guide](#)

Tips and FAQ

Is it Possible to Upload an Image from a Local Drive?

During feed processing, images must be stored on a non-password-protected web server so Amazon can retrieve them. Each image submitted must have a full URL, such as <http://mystore.com/images/1234.jpg>. Amazon cannot access images stored with a secured URL (https) so be sure to use http instead. Therefore, it is not possible to upload images from a local drive using the API directly.

As a possible solution, sign up for a free image bucket service and just upload your images there and supply the correct URLs Amazon needs (e.g. <http://mystore.com/images/1234.jpg>). After Amazon has the image, they store it locally so you can delete the image you have uploaded.
