



New Creators API Feeds Launch

AMAZON CONFIDENTIAL

Introduction

We're excited to announce the general availability of Creators API, featuring the new Feeds API functionality. The Feeds API, which includes ListFeeds and GetFeed endpoints, is now the standard method for data feeds access and is replacing the previous UI experience.

Creators API offers:

- Secure authentication through Amazon Cognito
- Enhanced performance and reliability
- Streamlined data access via ListFeeds and GetFeed APIs

Migration steps

Step 1: Generate credentials on Associates Central – Look for “CreatorsAPI” tab under Tools hamburger or click [here](#).

- Click Create Application – Enter any name
- Click Create Credential – Copy the values displayed on the screen

Note: Please use the store that's attached to your S3Proxy account to generate the credentials. If you don't know your Store ID, please contact your Business Development representative.

Step 2: Download the SDK for the language you want to integrate with

Step 3: Follow the SDK specific steps to integrate

SDKs

Node.js SDK : Link

1. Download & extract

- Extract the **creatorsapi-nodejs-sdk** zip file
- Navigate to the extracted directory

2. Install dependencies

- Run `npm install` to install all required dependencies and `npm run build` to build the SDK

3. Configure credentials & marketplace

- Open the project in any editor of your preference
- Navigate to the examples directory
- Add your **Credential ID**, **Credential Secret**, and **Version** in the sample files
- Set **marketplace** value (e.g., "www.amazon.com" for US marketplace) to which you want to send request
 - For marketplace values, please refer to: <https://webservices.amazon.com/paapi5/documentation/locale-reference.html>

4. Run sample code

- Execute samples like `node sampleListFeeds.js` to list the feeds
- Modify `sampleGetFeed.js` by entering the feed name (feed name can be found from the list feeds API response) you want to access and `run node sampleGetFeed.js`

Python SDK: Link

1. Download & extract

- Extract the **creatorsapi-python-sdk** zip file
- Navigate to the extracted directory

2. Install dependencies

- Install dependencies using `pip3 install -r requirements.txt`

3. Configure credentials & marketplace

- Open the project in any editor of your preference
- Navigate to the examples directory
- Add your **Credential ID**, **Credential Secret** and **Version** in the sample files
- Set **marketplace** value (e.g., "www.amazon.com" for US marketplace) to which you want to send request
 - For marketplace values, please refer to: <https://webservices.amazon.com/paapi5/documentation/locale-reference.html>

4. Run sample code

- Execute samples like `python3 sample_list_feeds.py` to list the feeds
- Modify `sample_get_feed.py` by entering the feed name (feed name can be found from the list feeds api response) you want to access and run `python3 sample_get_feed.py`

PHP SDK: Link

1. Download & extract

- Extract the **creatorsapi-php-sdk** zip file
- Navigate to the extracted directory

2. Install dependencies

- Install dependencies using `composer install`

3. Configure credentials & marketplace

- Open the project in any editor of your preference
- Navigate to the examples directory
- Add your **Credential ID**, **Credential Secret** and **Version** in the sample files
- Set **marketplace** value (e.g., "www.amazon.com" for US marketplace) to which you want to send request
 - For marketplace values, please reference [here](#)

4. Run sample code

- Execute samples like `php SampleListFeeds.php` to list the feeds
- Modify `SampleGetFeed.php` by entering the feed name (feed name can be found from the list feeds api response) you want to access and run `php SampleGetFeed.php`

Java SDK: Link

1. Download & extract

- Extract the **creatorsapi-java-sdk** zip file
- Navigate to the extracted directory

2. Configure credentials & marketplace

- Open the project in any editor of your preference
- Navigate to the examples directory
- Add your **Credential ID**, **Credential Secret** and **Version** in the sample files
- Set **marketplace** value (e.g., "www.amazon.com" for US marketplace) to which you want to send request
 - For marketplace values, please reference [here](#)

3. Run sample code

- Execute samples like `php SampleListFeeds.java` to list the feeds
- Modify `SampleGetFeed.java` by entering the feed name (feed name can be found from the list feeds api response) you want to access and run `SampleGetFeed.java`

Usage notes

1. Call ListFeeds to discover available feeds
2. Call GetFeed with desired feedName to obtain feed download URL
3. Use the URL to download the feed file before expiration (typically 60 minutes)

API references

ListFeeds API

Returns a list of all available catalog feeds along with metadata about each feed.

Endpoint

```
POST /catalog/v1/listFeeds
```

Request format

No input parameters required.

```
{}
```

Request format

	Field	Type	Description
1	feeds	Array	List of Feed objects

Feed object

	Field	Type	Description
1	feedName	String	Unique identifier of the feed
2	size	Long	Size of the feed file in bytes
3	md5	String	MD5 hash for verification
4	lastUpdated	String	Timestamp when feed was last updated

Example request

```
POST /catalog/v1/listFeeds
Content-Type: application/json
{}
```

Example response

```
{
  "feeds": [{
    "feedName": "us_standardized_wireless_retail_20250825.base.xml.gz",
    "size": 1874465394,
    "md5": "a1b2c3d4e5f6g7h8i9j0k1l2m3n4o5p6",
    "lastUpdated": "1756132638000"
  },
  {
    "feedName": "us_standardized_wireless_retail_20250825_1.delta.csv.gz",
    "size": 1637824,
    "md5": "p6o5n4m3l2k1j0i9h8g7f6e5d4c3b2a1",
    "lastUpdated": "1756138625000"
  }
]
```

GetFeed API

Provides a pre-signed S3 URL to download a specific feed file.

Endpoint

```
POST /catalog/v1/getFeed
```

Request format

	Parameter	Type	Required	Description
1	feedName	String	Yes	Name of the feed to retrieve

Response format

	Parameter	Type	Description
1	url	String	Pre-signed URL for downloading the feed file

Example request

```
POST /catalog/v1/getFeed
Content-Type: application/json

{
  "feedName": "us_standardized_wireless_retail_20250825.base.xml.gz"
}
```

Example response

```
{
  "url": "https://d1tmhd1o3zxb79.cloudfront.net/us_standardized_wireless_retail_20250825.base.xml.gz?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=..."
}
```

Error codes

Both APIs may return the following error responses:

	Error Type	HTTP Status Code	Description
1	ValidationException	400	Invalid request parameters or format
2	UnauthorizedException	401	Missing or bad authentication credentials
3	AccessDeniedException	403	Insufficient permissions to access the resource
4	ResourceNotFoundException	404	Requested feed not found
5	InternalServerErrorException	500	Unexpected server error during processing