AWS Data Exchange User Guide

AWS Data Exchange User Guide

Table of Contents

What Is AWS Data Exchange?

AWS Data Exchange is a service that makes it easy for AWS customers to securely exchange file-based data sets in the AWS Cloud.

As a subscriber, you can find and subscribe to hundreds of products from qualified data providers. Then, you can quickly download the data set or copy it to Amazon S3 for use across a variety of AWS analytics and machine learning services. Anyone with an AWS account can be a AWS Data Exchange subscriber. For information about becoming a subscriber, see Subscribing to Data Products on AWS Data Exchange (p. 7).

For providers, AWS Data Exchange eliminates the need to build and maintain any data delivery, entitlement, or billing technology. Providers in AWS Data Exchange have a secure, transparent, and reliable channel to reach AWS customers and grant existing customers their subscriptions more efficiently. The process for becoming an AWS Data Exchange provider requires a few steps to determine eligibility. For more information, see Register to Be a Provider (p. 14).

What Is An AWS Data Exchange Product?

A product is the unit of exchange in AWS Data Exchange that is published by a provider and made available for use to subscribers. When a provider publishes a product, that product is listed on AWS Data Exchange and AWS Marketplace. A product has the following parts:

- **Product details** – This information includes name, descriptions (both short and long), logo image, and support contact information. Providers complete the product details. For more information as a subscriber, see Product Subscriptions (p. 9). For more information as a provider, see Filling Out Product Details (p. 19).
- **Product offers** – To make a product available on AWS Data Exchange, providers must define a public offer. This offer includes prices and durations, data subscription agreement, refund policy, and the option to create custom offers. For more information, see Creating an Offer for AWS Data Exchange Products (p. 22).
- **Data sets** – A product can contain one or more data sets. A data set is a dynamic set of file-based content. Data sets are dynamic and are versioned through the use of revisions. Each revision can contain multiple assets. For more information, see Working with Data Sets (p. 28).

Malware Prevention

Security and compliance is a shared responsibility between you and AWS. To promote a safe, secure, and trustworthy service for everyone, AWS Data Exchange scans all data published by providers before it is made available to subscribers. If AWS detects malware, the affected asset is removed.

> **Important**
> AWS Data Exchange does not guarantee that the data you consume as a subscriber is free of any potential malware. We encourage that you conduct your own additional due-diligence to ensure compliance with your internal security controls. You can find anti-malware and security products in AWS Marketplace.

Supported Data Sets

AWS Data Exchange takes a responsible approach to facilitating data transactions by promoting transparency through use of the service. AWS Data Exchange reviews permitted data types, restricting

products that are not permitted. Providers are limited to distributing data sets that meet the legal eligibility requirements set forth in the Terms and Conditions for AWS Marketplace Sellers.

For more information about permitted data types, see Publishing Guidelines (p. 17).

Important
As an AWS customer, you are encouraged to conduct your own additional due-diligence to ensure compliance with any applicable data privacy laws. If you suspect that a product or other resources on AWS Data Exchange are being used for abusive or illegal purposes, report it using the Report Amazon AWS abuse form.

Pricing

For pricing information, see http://aws.amazon.com/data-exchange/pricing.

Supported Regions

AWS Data Exchange has a single, globally available product catalog offered by providers. Subscribers can see the same catalog regardless of which AWS Region they are using. The resources underlying the product (data sets, revisions, assets) are regional resources that you manage programmatically or through the AWS Data Exchange console in supported AWS Regions. For information about which regions are supported, see Global Infrastructure Region Table.

Related Services

The following services are related to AWS Data Exchange:

- **Amazon S3** – Currently, the only supported asset type for data sets is Amazon S3 object snapshots. Subscribers can export data sets to Amazon S3 programatically. For more information, see What Is Amazon S3? in the *Amazon Simple Storage Service Developer Guide*.
- **AWS Marketplace** – AWS Data Exchange allows data sets to be published as products on AWS Marketplace. AWS Data Exchange providers must be registered as AWS Marketplace sellers, and can use the AWS Marketplace Management Portal or the AWS Marketplace Catalog API. For information about becoming an AWS Marketplace subscriber, see What Is AWS Marketplace? in the *AWS Marketplace Buyer Guide*. For information about becoming an AWS Marketplace seller, see What Is AWS Marketplace? in the *AWS Marketplace Seller Guide*.

How AWS Data Exchange Works

With AWS Data Exchange, providers publish file-based data products and subscribers subscribe to those products.

How to Subscribe to Products

At a high level, this is how to use AWS Data Exchange as a subscriber.

1. **Potential subscribers browse the catalog** – Products are published on AWS Data Exchange and are also available on AWS Marketplace. You can find products and review the associated public or custom offers and product details. For more information, see Subscribing to Data Products on AWS Data Exchange (p. 7).
2. **(Optional) Potential subscribers submit a request for a subscription** – The provider can choose to enable subscription verification. If they do so, you must request a subscription to the product. For more information, see Subscription Verification for Subscribers (p. 10).
3. **Subscriber subscribes to the product** – If you subscribe to a paid product, you are billed on your AWS bill. You get access to the entitled data set. For more information, see Subscribing to Data Products on AWS Data Exchange (p. 7).
4. **Subscriber uses the product** – You have access to the product data sets according to the terms of the data subscription agreement. You can export the associated assets to Amazon S3 or you can use jobs with a signed URL. For more information, see Jobs (p. 32).
5. **Request new data products** – If you are not able to find a product in the catalog, you can use the **Request data product page** in the Console to inform AWS of your interest. AWS will use this information to work with the data provider, and try to get that data added to the catalog.

How to Provide Data Products

At a high level, this is how to use AWS Data Exchange as a provider.

1. **Potential provider registers to be a provider** – Registering allows you to list products on AWS Data Exchange and make them available on AWS Marketplace. For more information, see Register to Be a Provider (p. 14).
2. **The data is eligible to be published on AWS Data Exchange** – You're limited to distributing data sets that meet the legal eligibility requirements set forth in the Terms and Conditions for AWS Marketplace Sellers. For more information about the types of permitted data, see Publishing Guidelines (p. 17).
3. **Provider creates a data set and imports assets** – You can use your files or Amazon S3 objects to create data sets through the AWS Data Exchange console or APIs. Then, you can create revisions in the data set, and import assets into that revision. Assets can be imported from either Amazon S3 or through the use of a signed URL using asynchronous workflows called jobs. For more information, see Working with Data Sets (p. 28).
4. **Provider creates a product and its public offer** – To create a product, you must provide product details, include one or more data sets, and provide public offer details. For more information, see Publish a Product (p. 16).
5. **AWS Data Exchange copies the data set** – When an owned data set is published in a product, AWS Data Exchange creates a copy of the data set. Subscribers can access that copy of the data set as an entitled data set.

6. **(Optional) Provider enables subscription verification** – If you enable subscription verification, subscribers must request a subscription to your product. This gives you an opportunity to review potential subscribers before they access your data sets. For more information, see Subscription Verification for Providers (p. 25).
7. **(Optional) Provider creates custom offers for the product** – In addition to the public offer, you can create custom offers, including private and BYOS offers, for select customers. For more information, see Creating Custom Offers (p. 23).
8. **(Optional) Provider publishes new revision** – You can update dynamic data sets over time by creating a new revision using the AWS Data Exchange APIs or console. These revisions can then be published. For more information, see Revisions (p. 30) or Updating Products (p. 21).
9. **Provider reviews reports through the AWS Marketplace Management Portal** – Reports are available to all registered AWS Marketplace seller and are released on a regular cadence (daily, weekly, or monthly). For more information, see Provider Financials on AWS Marketplace (p. 26).
10. **Provider receives funds distributed by AWS Marketplace** – For more information, see Provider Financials on AWS Marketplace (p. 26).

Programmatic Access

If you're using AWS Data Exchange programmatically, there are two different sets of resources with two different APIs.

- **AWS Data Exchange APIs** – Use these APIs to create, view, update, and delete data sets and revisions. You can also use these APIs to import and export assets to and from those revisions. For more information, see the AWS Data Exchange API Reference.
- **AWS Marketplace Catalog APIs** – Used by providers to view and update products on AWS Data Exchange and AWS Marketplace. For more information, see the AWS Marketplace Catalog API Reference.

Setting Up AWS Data Exchange

Before you can use any AWS service, including AWS Data Exchange, you must complete the following tasks:

Topics

Sign Up for AWS

If you do not have an AWS account, complete the following steps to create one.

To sign up for an AWS account

1. Open https://portal.aws.amazon.com/billing/signup.
2. Follow the online instructions.

   Part of the sign-up procedure involves receiving a phone call and entering a verification code on the phone keypad.

Create an IAM User

To create an administrator user for yourself and add the user to an administrators group (console)

1. Sign in to the IAM console as the account owner by choosing **Root user** and entering your AWS account email address. On the next page, enter your password.

   Note
   We strongly recommend that you adhere to the best practice of using the **`Administrator`** IAM user below and securely lock away the root user credentials. Sign in as the root user only to perform a few account and service management tasks.
2. In the navigation pane, choose **Users** and then choose **Add user**.
3. For **User name**, enter **`Administrator`**.
4. Select the check box next to **AWS Management Console access**. Then select **Custom password**, and then enter your new password in the text box.
5. (Optional) By default, AWS requires the new user to create a new password when first signing in. You can clear the check box next to **User must create a new password at next sign-in** to allow the new user to reset their password after they sign in.
6. Choose **Next: Permissions**.
7. Under **Set permissions**, choose **Add user to group**.
8. Choose **Create group**.
9. In the **Create group** dialog box, for **Group name** enter **`Administrators`**.
10. Choose **Filter policies**, and then select **AWS managed -job function** to filter the table contents.
11. In the policy list, select the check box for **AdministratorAccess**. Then choose **Create group**.

Note
You must activate IAM user and role access to Billing before you can use the `AdministratorAccess` permissions to access the AWS Billing and Cost Management console. To do this, follow the instructions in step 1 of the tutorial about delegating access to the billing console.

12. Back in the list of groups, select the check box for your new group. Choose **Refresh** if necessary to see the group in the list.
13. Choose **Next: Tags**.
14. (Optional) Add metadata to the user by attaching tags as key-value pairs. For more information about using tags in IAM, see Tagging IAM Entities in the *IAM User Guide*.
15. Choose **Next: Review** to see the list of group memberships to be added to the new user. When you are ready to proceed, choose **Create user**.

You can use this same process to create more groups and users and to give your users access to your AWS account resources. To learn about using policies that restrict user permissions to specific AWS resources, see Access Management and Example Policies.

Note
Only AWS accounts that are registered to provide data products on AWS Marketplace and AWS Data Exchange can create data sets and products.

Subscribing to Data Products on AWS Data Exchange

Before you subscribe to a product on AWS Data Exchange, review the following topics to understand how to be a subscriber and how to get set up to use AWS and AWS Data Exchange:

- How to Subscribe to Products (p. 3)
- Setting Up AWS Data Exchange (p. 5)

Note
When subscribing to data products from some non-US sellers, you might also receive a tax invoice from the seller. For more information, see Tax Help - AWS Marketplace Sellers.

Getting Started as a Subscriber

AWS Data Exchange Heartbeat is a free product offered by AWS Data Exchange that you can use to subscribe to product. You can also use it for testing and to familiarize yourself with how to use AWS Data Exchange and its concepts. For more information, see What Is AWS Data Exchange Heartbeat? (p. 8)

Subscribe to AWS Data Exchange Heartbeat on AWS Data Exchange

Use this procedure to browse the AWS Data Exchange catalog to find listings of data sets.

To find and subscribe to AWS Data Exchange Heartbeat

1. Open the AWS Data Exchange console.
2. From the left navigation pane, choose **Product catalog**.
3. From the search bar, type **`AWS Data Exchange Heartbeat`** and press **Enter**. Choose the product to view its details page.

   The information on the details page includes a product description, the provider's contact information, and the details of the product's public offer. The public offer information includes price and durations, the data subscription agreement, and the refund policy. You can also view the names of the data sets included in the product and the AWS Regions in which they are available.

   If the provider has issued a custom offer to your account (for example, a private offer or BYOS offer), you see those details, too.
4. In the top right corner, choose **Continue to Subscribe**. From here, you can choose your preferred price and duration combination, choose whether to enable auto-renewal for the subscription, and review the offer details, including the data subscription agreement.

   Note
   AWS Data Exchange Heartbeat doesn't require subscription verification, but some products do. For more information, see Subscription Verification for Subscribers (p. 10).
5. Review the pricing information, choose the pricing offer, and then choose **Subscribe**.

Note
AWS Data Exchange Heartbeat is a free product. If you subscribe to a paid product, you are prompted to confirm your decision to subscribe.

It might take several minutes for the subscription to be processed. Then, you should see it on your **Subscriptions** page.

6. After you subscription is active, you can now access the data sets for AWS Data Exchange Heartbeat.
7. To view your subscriptions, from the left navigation pane, choose **Subscriptions**, and then choose **AWS Data Exchange Heartbeat (Test product)**. The data sets that are part of the product are displayed. You can enable or disable auto-renewal for your subscription on this page.
8. When you choose the **AWS Data Exchange Heartbeat** data set, you can view the data set's ID, name, and description. For more information, see Working with Data Sets (p. 28).
9. On the **Revisions** tab, you can view the data set's revisions, from latest to oldest. To view the details of a revision, choose its revision ID.

   The revision's details include its assets, displayed in a table. To export one or more assets, select the check boxes, and then choose **Export to Amazon S3** or **Download**.

Related Topics

- What Is AWS Data Exchange Heartbeat? (p. 8)
- Subscription Verification for Subscribers (p. 10)
- Accepting a Private Offer (p. 11)
- Accepting a Bring Your Own Subscription Offer (p. 11)
- Working with Data Sets (p. 28)

What Is AWS Data Exchange Heartbeat?

AWS Data Exchange Heartbeat (Test product) is a free product that is made available to subscribers to understand how to interact with a AWS Data Exchange product subscription. You can use it for testing purposes and to get familiar with the AWS Data Exchange APIs and concepts.

AWS Data Exchange Heartbeat contains a single data set named **Heartbeat**. Approximately every 15 minutes, a new revision is published to this data set.

Example Content of a Revision

Each new revision contains two assets:

- Epoch asset
- Manifest asset

Epoch Asset

Each AWS Data Exchange Heartbeat revision contains a JSON file S3 object that contains a single array. The array's name is `TimestampsSinceLastRevision` and its value is a list of each UNIX Epoch second that has elapsed since the last revision.

The name of the asset is in the form `Epoch{start}-{end}.json` where `{start}` and `{end}` represent the Epoch seconds corresponding to the period of time covered by the revision.

Manifest Asset

Each AWS Data Exchange Heartbeat revision contains a JSON file S3 object that contains metadata about the revision and the schema of the Epoch asset JSON file. The name of the asset is in the form `Manifest{start}-{end}.json` where `{start}` and `{end}` represent the Epoch seconds corresponding to the period of time covered by the revision. This example shows the content of a manifest file:

```
{
        "manifestSchemaVersion":"1.0",
        "schema":"{
                \"type\":\"object\",
                \"properties\":{
                    \"TimestampsSinceLastRevision\":{
                        \"type\":\"array\",
                        \"description\":\"List of epoch timestamps in seconds.\",
                        \"items\":{
                            \"type\":\"number\",
                            \"description\":\"Epoch timestamp in seconds.\"
                         }
                     }
                 }
        }",
        "startTimestamp":1554898111,
        "endTimestamp":1554905311,
        "numberOfTimestamps":7201
}
```

Product Subscriptions

All AWS Data Exchange products are subscription-based. When you subscribe to a product, you agree to the product's offer terms, including the price, duration, data subscription agreement, and refund policy. When you subscribe to a product, you pay any associated charges upfront for the duration that you subscribed to.

> **Important**
> The data subscription agreement (DSA) sets forth the provider's terms and conditions for the data product. Use of any data product subscribed to on AWS Data Exchange must also be in compliance with the AWS Customer Agreement or other agreement governing your use of AWS services.

Each product's offer terms can contain one or more price and duration combinations. When you subscribe to a product, you can choose the duration of the subscription. You can also choose whether you would like to enable auto-renewal for that subscription.

After a subscription is processed and active, it appears on your AWS bill as part of your AWS Marketplace charges. For more information, see AWS Marketplace Paying for Products.

During the duration of your subscription, you can view and access all the product's data sets. You can also export the data sets' assets in jobs. For more information, see Jobs (p. 32). Once a subscription has expired, you can no longer view or export the data sets.

> **Note**
> When you subscribe to a product, you can access all of the data set revisions published to that product, regardless of when they were published.

A product subscription gives you access to all the data set's revisions that have been published to the product you are subscribed to. If a provider decides to unpublish a product, you still have access to the

data sets as long as your subscription is active. However, you cannot auto-renew the subscription when it expires.

You can view all of your active product subscriptions and auto-renewal status on the **Subscriptions** page of the AWS Data Exchange console. Visit the **Entitled data sets** page to find and access all of your entitled data sets in a specific AWS region, based on your active subscriptions.

> **Important**
> If you enable auto-renew, and the product's offer terms have changed at the time of renewal, then the new product offer terms (including new price and new DSA) apply. This ensures that you keep access to the data regardless of potential offer terms changes.

When you subscribe to a data product, we might share your contact information with the provider. For more information, see What Information Do You Share with the Software Seller about the Customers of a Product?.

Subscription Verification for Subscribers

For various reasons, including compliance or regulatory reasons, some data providers might choose to restrict access to their products using subscription verification. When you subscribe to these data products, you are required to submit additional information about who you are and your intended use-case. The provider reviews this information before approving subscriptions.

For products that require subscription verification, when you choose **Continue** to subscribe on a product page, a subscription request page appears. You must provide the following information:

- Your company name
- Your name
- Your email address
- Your intended use case for the data product, along with any other comments that the provider might find useful when reviewing the subscription request
- Your AWS account ID (added automatically)

After you submit your request, the provider has up to 45 days to approve or decline your request. To review your pending subscription requests, sign in the AWS Management Console and open the AWS Data Exchange console. Choose **Subscriptions**, and then choose **Subscription requests**. After a provider approves your request, the subscription appears on the **Subscriptions** page.

Each subscription request is uniquely identified by its ID. The ID is visible to both the provider and the subscriber. You can use the subscription request ID to identify the request in your communications with the provider.

> **Note**
> You can cancel a pending subscription request at any time as long as it hasn't expired or already been processed.

Email Notifications

You will receive an email notification to your AWS account's email address when your request is approved, declined, or when it expires. Although most subscription request status changes result in an email notification, the delivery of these emails is on a best-effort basis.

> **Note**
> You will not receive email notifications for subscription request status changes that you have initiated yourself (for example, cancelling a subscription).

Accepting a Bring Your Own Subscription Offer

As a subscriber, you might want to migrate your existing data subscriptions to AWS Data Exchange. Bring your own subscription (BYOS) functionality allows you to migrate and fulfill existing subscriptions with participating data providers at no additional cost.

With BYOS offers, any billing relationship between providers and subscribers continues. BYOS offers are not subject to fulfillment fees. As a subscriber, you receive an AWS Marketplace invoice for the subscription with no charge.

Because the subscription lifecycle starts outside of AWS Data Exchange, the workflow for migrating the existing subscriptions to AWS Data Exchange using BYOS requires collaboration between the provider and subscriber.

> **Important**
> With BYOS offers, you're migrating a subscription that predates the availability of this product on AWS. AWS might verify your BYOS offer with the existing subscription agreement. If AWS cannot verify your BYOS offer, the offer and entitlements can be revoked without notice.

Before creating or accepting a BYOS offer on AWS Data Exchange, the provider and subscriber should perform the following steps together:

Prerequisites

1. The provider and the subscriber should contact each other about implementing a BYOS AWS Data Exchange solution.
2. The subscriber provides the AWS account ID that they want to use to subscribe to data products on AWS Data Exchange.

The subscriber accepts the BYOS offer as follows.

To accept a BYOS offer

1. Sign in to the AWS Data Exchange console.
2. In the left navigation pane, from **Discover data**, choose **Product catalog**.
3. In **Product catalog**, search for the name of the product or paste the product URL that the provider gave you (for example, AWS Data Exchange Heartbeat).
4. Choose the name of the product to review the BYOS offer.
5. Choose **Continue to subscribe**.
6. Review the terms of the offer, the data subscription agreement, and the included data sets.
7. If you accept the terms of the offer, review and accept the acknowledgement, and then choose **Subscribe**.

Accepting a Private Offer

Data providers can provide their data product to you at terms that are different from the offer terms available to the general public. A private offer can be different from the public offer in any dimension, including price, duration, data subscription agreement, or refund policy.

> **Note**
> Unlike BYOS offers, private offers are not required to be based on an existing subscription that predates the product's availability on AWS Data Exchange.

The provider must first create a public offer and then a custom offer for your AWS account. If a private offer hasn't been extended to you, you can request one by contacting a provider using the contact information on the details page of the public offer.

As a subscriber, you can accept a private offer as follows.

To accept a private offer

1. Sign in to the AWS Data Exchange console.
2. In the left navigation pane, from **Discover data**, choose **Product catalog**.
3. In **Product catalog**, search for the name of the product or paste in the product URL (for example, AWS Data Exchange Heartbeat).
4. Choose name of the product to review the private offer.
5. Choose **Continue to subscribe**.
6. Review the terms of the offer, the data subscription agreement, and the included data sets.
7. If you accept the terms of the offer, review and accept the acknowledgement, and then choose **Subscribe**.

Providing Data Products on AWS Data Exchange

Before you become a data product provider on AWS Data Exchange, review the following topics:

- How to Provide Data Products (p. 3)
- Setting Up AWS Data Exchange (p. 5)

After you review these topics, you're ready to get started.

Getting Started as a Provider

These topics describe the end-to-end process of becoming a data product provider on AWS Data Exchange using the AWS Data Exchange console. The process has the following steps:

Topics

Confirm Your Eligibility

Before you can register, you must meet the following requirements to confirm your eligibility.

Seller requirements for publishing data products

Whether you charge for your AWS Data Exchange data product or not you're selling that product on AWS Marketplace. To create and offer data products you must:

- Have a defined customer support process and support organization.
- Provide a means to keep data regularly updated and free of vulnerabilities.
- Follow best practices and guidelines when marketing your product.
- Be an AWS customer in good standing and meet the requirements in the terms and conditions for AWS Marketplace sellers and for AWS Data Exchange providers.
- You must be a permanent resident or citizen in an eligible jurisdiction (p. 14), or a business entity organized or incorporated in one of those areas.

- You must provide tax and bank account information. For US-based entities, a W-9 form and a banking account from a US-based bank are required.
- Non-US sellers are required to provide a (i) W-8 form, value-added tax (VAT) or goods and services tax (GST) registration number, and (ii) US bank information. If you don't have a US bank account, you can register for a virtual US bank account from Hyperwallet.
- To provide data products, you must also request on-boarding through the Create case wizard for AWS Support. The AWS Data Exchange team will contact you to complete the qualification and registration process.

Eligible jurisdictions for AWS Data Exchange products

To provide data products on AWS Data Exchange, you must be a permanent resident or citizen in one of the following countries, or a business entity organized or incorporated therein:

- Australia[1]
- Bahrain[1][2]
- European Union (EU) member state[1]
- New Zealand[1]
- Norway[1][2]
- Switzerland[1][2]
- United Arab Emirates (UAE)[1][2]
- United Kingdom (UK)[1]
- United States (US)

[1] Sellers of paid products in these countries must provide VAT information. For more information, see Tax Help - AWS Marketplace Sellers.

[2] In these countries, sellers may need to provide an invoice to buyers. For more informations, see Tax Help - AWS Marketplace Sellers.

Register to Be a Provider

To use AWS Data Exchange as a provider, you must be a registered seller on AWS Marketplace and be qualified by the AWS Data Exchange team. When you register an account as an AWS Marketplace seller, the account is the seller of record for your products and is used for reporting and disbursement. All products and their public offers are discoverable on AWS Data Exchange and AWS Marketplace.

> **Important**
> You cannot change the AWS account you use to list a product on AWS Marketplace. Only data sets owned by that account can be included in products published by that account. Only AWS accounts that are registered to provide data products on AWS Marketplace and AWS Data Exchange can publish products.

To register as a provider for AWS Data Exchange and AWS Marketplace

1. From your web browser, open the AWS Marketplace Management Portal.
2. Choose **Sign Up as an AWS Marketplace Seller** to open the registration wizard.
3. Confirm your company or full name, and review the Terms and Conditions. If you agree to them, choose **I have read and agree to these terms**.
4. On the **Account Settings** page, from the **Provide tax and banking information** tab, choose **Start** to complete the tax and banking wizard. This submits your tax and banking information in the AWS Marketplace Management Portal.

Note
We strongly recommend that you sign and submit the tax form electronically. Otherwise, you must print, complete the signature section, and mail a hard copy of the tax form to the address provided in the tax information interview. This delays the registration process.

5. On the **Account Settings** page, choose **Add** to add a public profile. Your registration for the AWS Marketplace is now complete.

   Note
   While your tax and banking registration is pending, you cannot submit a public profile.

6. In addition to being a registered AWS Marketplace seller, you must submit an AWS Data Exchange qualification request. Access the AWS Support Dashboard and create a case in the AWS Management Console. The AWS Data Exchange team will contact you to complete the qualification and registration process.

Confirm Eligibility of Your Data

You're limited to distributing data sets that meet the legal eligibility requirements set forth in the Terms and Conditions for AWS Marketplace Sellers. If a provider breach these terms in any way, the prohibited product is removed from AWS Data Exchange and the provider might be suspended from the service. For more information, see Publishing Guidelines (p. 17).

If you have questions about the eligibility of your data set, access the AWS Support Dashboard and create a case in the AWS Management Console. After you've reviewed the publishing guidelines for data products on AWS Data Exchange, and you've confirmed that your data set can be listed, you create your product.

Create a Data Set

In the following procedure, you configure a data set, create a revision, add data assets, and prepare it for publishing in the AWS Data Exchange console. Data sets are dynamic and are versioned using revisions, with each revision containing at least one asset. For more information, see Working with Data Sets (p. 28).

It's assumed you have already created files for your data sets and stored them as objects in Amazon S3 or on your local computer. AWS Data Exchange supports all file types.

To create a data set

1. Open your web browser and go to the AWS Data Exchange console.
2. In the left side navigation pane, from **Publish data**, choose **Data sets**.
3. In **Data sets**, choose **Create data set** to open the wizard.
4. Enter a name and description for your data set, and then choose **Create**. You can also add tags to your data sets. For more information, see Data Set Best Practices (p. 30).
5. Edit or deleted information about your data set, and then choose **Create revision**.
6. Provide an optional comment for your revision that describes the purpose of the revision, and then choose **Create**. You can also add tags for your revision.
7. Again, you can review, edit, or delete your changes from the previous step. After that, expand **Import assets**, and choose either **from Amazon S3** or **from your computer**, depending on where the data assets for the data set are currently stored.
8. This starts a job to import your asset into your data set. After the job is finished, the **State** field in the **Jobs** section is updated to **Completed.**
9. If you have more data to add, choose **Import assets** to add assets to this revision.

10. Review your revision and its assets. If it's complete, choose **Finalize** to stage it for publishing.

You have successfully finalized a revision for a data set. It's now ready to be published as a part of a product.

Publish a Product

After you've created at least one data set and finalized a revision with assets, you're ready to publish that data set as a part of a product. For more information, see Publishing Products (p. 18). Make sure that you have all required details about your product and offer.

To publish a product

1. Open your web browser and go to the AWS Data Exchange console.
2. From the left navigation pane, expand **Publish data**, and choose **Products dashboard**.
3. From **Products**, choose **Publish new product** to open the wizard.
4. Enter information about your product, including name, logo, support contact, web address, categories, and descriptions. For more information, see Filling Out Product Details (p. 19).
5. Review your information, and then choose **Next**.
6. Select the check box next to the data sets you want to add.

   Note
   The data sets you choose must have a finalized revision. Data sets without finalized revisions won't be added.
7. Choose **Add selected**, and then scroll to **Selected data sets**.
8. Review your data sets, and then choose **Next**.
9. Configure your public offer. All AWS Data Exchange products require a public offer. Choose your price and subscription durations, U.S. sales tax settings, data subscription agreement, and refund policy. For more information, see Creating an Offer for AWS Data Exchange Products (p. 22).
10. (Optional) Set **Subscription verification**, which enables you to control who can subscribe to this product. For more information, see Subscription Verification for Providers (p. 25).
11. Review your product information before you publish.
12. If you are sure you want sure to make the product and public offer visible and available to everyone, choose **Publish**.
13. You've now completed the manual portion of publishing a data product with a public offer. AWS Data Exchange prepares and publishes your product. On the **Product overview** page, the status of your product is **Publishing**.

Unpublish a Product

After your product is published, it's available for all to find and subscribe to. Use the following procedure if you created a product for this getting started exercise or if you'd like to clean up your resources. You should also follow the steps in this procedure to remove a product from the publicly listed products on AWS Data Exchange.

Keep the following in mind when you unpublish a product:

- You can unpublish a product whenever you want.
- If you unpublish a product, it is no longer visible in the AWS Data Exchange catalog or on AWS Marketplace.

- Subscribers with an active subscription maintain access to the data product until the term of their subscription expires.
- Active subscriptions that expire after you have unpublished your product are not renewed, even if the subscriber has enabled auto-renewal.
- Existing subscribers can still view the product details until their subscription expires.

To unpublish a product

1. Open your web browser and go to the AWS Data Exchange console.
2. From the left navigation pane, expand **Publish data**, and then choose **Products dashboard**.
3. From **Products**, choose the product you want to remove. Make sure its status is **Published**.
4. From **Product overview**, choose **Unpublish**, and then follow the instructions to unpublish the product.

   Important
   This action can't be undone.

After you complete these steps, your product's status is **Unpublished**. An unpublished product can't be published again, but you can create a new product (with a new product ID) that has the same data sets, product details, and offer details.

View Reports

AWS Data Exchange is integrated with AWS Marketplace, so you benefit from AWS Marketplace features, such as seller reports and the AWS Marketplace Commerce Analytics Service. For more information, see Provider Financials on AWS Marketplace (p. 26).

Related Topics

- Publishing Guidelines (p. 17)
- Publishing Products (p. 18)
- Creating an Offer for AWS Data Exchange Products (p. 22)
- Working with Data Sets (p. 28)

Publishing Guidelines

The following guidelines outline restrictions for listing products on AWS Data Exchange. As a provider, you are responsible for complying with these guidelines and the Terms and Conditions for AWS Marketplace Sellers. AWS may update these guidelines from time to time. AWS removes any product that breaches these guidelines and may suspend the provider from future use of the service.

AWS Data Exchange Publishing Guidelines for Data Products

1. Your data products may not contain any illegal content, viruses, malware, or any other material that is harmful to others.
2. Your data products may not include information that can be used to identify any person, unless that information is already legally available to the public. Permitted examples include newspaper articles, open court records, public company filings, or public online profiles.
3. The following categories of information must be aggregated or anonymized so that no person in your data product can be identified: biometric or genetic data, health, racial or ethnic origin, political

opinions, religious or philosophical beliefs, sex or sexual orientation, trade union membership, personal payment or financial information (for example, credit history), or other similar categories of sensitive information.

Some examples of data sets that can be included on AWS Data Exchange: (1) Historic stock prices for public companies, (2) Names of judges and their court opinions, and (3) Aggregated or anonymized research findings from pharmaceutical drug studies.

Some examples of data sets that are prohibited on AWS Data Exchange: (1) Lists of names organized by race, (2) Geo-location data that can be used to identify a person, and (3) protected health information under HIPAA.

4. You should carefully consider how subscribers may and may not use your data products, and you should clearly include this information in your Data Subscription Agreement (DSA).
5. Product listing descriptions must be accurate, contain valid contact information, and note if any data has been aggregated or anonymized.
6. You may not use AWS Data Exchange to promote any other products or solutions not listed on AWS Marketplace, except for products or solutions that are not compatible with AWS Marketplace.

If you have questions about the eligibility of your data set, contact AWS Support. After you've reviewed the publishing guidelines for data products on AWS Data Exchange, and you've confirmed that your data set can be listed, you can create your product.

Publishing Products

A product is the unit of exchange in AWS Data Exchange that is published by a provider and made available for use to subscribers. When you publish a product, that product is available on the AWS Data Exchange product catalog as well as the AWS Marketplace. Each product you publish is uniquely identified by its product ID.

Note
When a product is initially created and published, all pre-existing finalized revisions within its data sets are published at the same time.

You can publish and view your products using the AWS Data Exchange console. You can also list and view the details of your existing products using the AWS Marketplace Catalog API. A product has the following parts:

- **Product details** – This information helps potential subscribers understand what the product is. This includes a name, descriptions (both short and long), a logo image, and contact information. For more information, see Filling Out Product Details (p. 19).
- **Product offers** – In order to make a product available on AWS Data Exchange, you must define a public offer. This includes the price, data subscription agreement, refund policy, and more. Offers define the terms that subscribers are agreeing to when they subscribe to a product. Each product must have a public offer available to all subscribers. Providers can also create custom offers to specific subscribers. For more information, see Creating an Offer for AWS Data Exchange Products (p. 22).
- **Data sets** – A product can contain one or more data sets. A data set is a dynamic set of file-based data content. As a provider, you create owned data sets, and a subscriber can get access to entitled data sets through a product subscription. Data sets are dynamic and are versioned using revisions. You can decide which revisions within a data set are published to a product. For more information, see Updating Products (p. 21).

Note
When a subscriber subscribes to your product, they get access to the product's data sets and all the revisions that have been published to that product for the duration of their subscription.

Subscribers can also access revisions that have been published before their subscription became active.

Sensitive Information

Sensitive information is biometric or genetic data; health data; racial or ethnic origin; political opinions; religious or philosophical beliefs; sex or sexual orientation; trade union membership; personal payment or financial information (for example, credit history); or other similar categories of information.

If your product contains sensitive information, you must acknowledge that it cannot be used to identify a person. You make this acknowledgement on **Step 2: Add data** of the product creation wizard. If the product doesn't contain sensitive information, choose **No**. If the selection is **Yes**, then you must ensure that the sensitive data in your product is properly anonymized, de-identified, or aggregated such that the information is no longer identifying for any individuals.

Additionally, choosing **Yes** will result in the display of a message on this product's page on AWS Data Exchange. The message informs potential customers that the product contains sensitive information that has been appropriately anonymized, de-identified, or aggregated.

Warning
Listing a product with sensitive information that has not been anonymized, de-identified, or aggregated, or incorrectly acknowledging the state of sensitive data in your product, is a violation of our Publishing Guidelines (p. 17). AWS removes any product that breaches these guidelines and can suspend the provider from future use of the service.

Filling Out Product Details

When you publish a product on the AWS Data Exchange console, you must provide the product's details. This section covers some best practices to consider when you're preparing product details.

Product Name

Subscribers will search for the names of products, so make your product name something meaningful.

Short Description

The product short description text appears on the tiles in the product catalog portion of the AWS Data Exchange console. We recommend that you provide a concise description of your product for this field.

Long Description

Subscribers see the product long description in the product detail page after the product is published. We recommend that you list the product's features, benefits, usage, and other information specific to the product.

Product information in the description must accurately represent the data being provided to subscribers. This includes data coverage (for example, 30,000 financial instruments or 10,000 location coordinates) and data set update frequency (for example, daily updates or weekly updates).

Provide Additional Information

In order to make your product description compelling to prospective subscribers, we recommend you add the following information to your product description:

- *Data due diligence questionnaire (DDQ)*: Typically includes responses to questions regarding the firm selling a data set. Examples of the information in a DDQ includes the process that a provider

goes through to collect the data, or quality control procedures and questions regarding regulatory compliance.

- *Data set schemas*: Provide prospective users with detailed descriptions of the structure and format of your data sets. Examples of the information in a data set schema include the identification of a primary key, field names, field definitions, expected output types for each field (for example, string, integer), and acceptable enumerations for each field (for example, 0% - 100%).
- *Trial Product Listings*: Many prospective subscribers request trials of data sets before paying for a subscription. Trial products can be published on AWS Data Exchange for subscribers to subscribe to like regular paid products.
- *Sample files*: Sample files are typically smaller versions, or older, out-of-date versions of full production data sets. These sample files give prospective users insights into the outputs they can expect before purchasing a subscription.
- *Product fact sheets*: These can be documents, web links, or both to provide subscribers with more granular statistics on the coverage of your data sets, typical use-cases for your data sets, and any other factors that differentiate your data sets.

For information about adding links in the description, see Include Links in Your Product Description (p. 20).

Product Logo

The product logo appears in the AWS Data Exchange product catalog on the console and on AWS Marketplace. The supported formats for the logo are .png, .jpg, and .jpeg.

Support Contact

As a provider, you must include valid contact information. This can be a managed email alias or case management system link for customers to use to get help when they have questions about your product. We strongly recommend that you don't use a personal email address because the address is publicly visible.

Product Categories

All products fit into one or more categories. By specifying up to two categories for your product, you help subscribers filter and find your products in AWS Data Exchange and AWS Marketplace.

Include Links in Your Product Description

The long description for a AWS Data Exchange product supports Markdown, which allows you to include links in your product's details page. The following procedure shows you how to add links to websites in your AWS Data Exchange product description. Complete the following steps.

To include embedded links in your product listing

1. Log into the AWS console and navigate to a Amazon S3 bucket that your AWS Data Exchange user account has access to. The contents of this bucket are publicly readable.
2. Upload the files (for example, documents such as PDF files or Microsoft Excel files) that you want to include in your product listing into the Amazon S3 bucket. After the upload is complete, make sure you set the file or files to have public read access permissions.
3. Choose one of the uploaded files. In the **Overview** tab, you will see a URL for the file. Copy the URL to your clipboard.
4. Open the AWS Data Exchange console at AWS Data Exchange console.

5. Choose the product you want to update, and then choose **Edit**.
6. From **Product Description**, use the following Markdown formats to link to relevant files (using the URL link you copied previously) or to another URL, like your website.

   - To link to a file stored in an Amazon S3 bucket:

     ```
     **_[File name](Object URL from Amazon S3)_**

     Description of the object.
     ```

   - To link to a trial product listing on AWS Data Exchange:

     ```
     **_[Website Title](URL)_**

     Description of the website.
     ```

7. Choose **Save Changes**. After a few minutes your AWS Data Exchange product listing page should be updated with the new links.

Updating Products

These sections describe how to update your products. The instructions are written with the assumption you're a provider who's familiar with Working with Data Sets (p. 28). After you publish a product, you can edit the product's details and its public offer. You can also update the underlying data sets by publishing new revisions to subscribers.

Update Product and Offer Details

After you publish a product, you can use the AWS Data Exchange console to edit the product details. You can also edit the product's public or custom offers and change the offer terms. When you update your product's offer terms, subscribers with an active subscription keep their existing offer terms as long as their subscription is active. Subscribers who have chosen auto-renewals use the new offer terms.

Keep the following in mind when you update products:

- You can't remove or edit a subscription duration in your offers. This ensures that existing subscribers retain the ability to renew. If you no longer want to offer a specific subscription duration, you can un-publish your existing product and then publish a new product. For more information, see Unpublish a Product (p. 16).
- You can't add or remove data sets from a product after it is published, regardless of how many subscribers have subscribed to your product.

Publish New Data to Products using Revisions

AWS Data Exchange supports dynamically updated products. Subscribers subscribe to the product for a certain duration and access all of the published data set revisions as long as their subscription is active. For example, a provider might want to provide a product that contains daily closing stock prices for U.S equities, which would be updated every day with the day's closing prices.

You can use the AWS Data Exchange console or the AWS Marketplace Catalog API to publish new revisions to a product. For more information, see Using AWS Data Exchange with the AWS Marketplace Catalog API (p. 56).

> **Important**
> Any revision published to a product is immutable and can't be edited, changed, or deleted. If you need to remove published content for compliance reasons, contact AWS Support.

Suggested Approach for Historical Data

Some dynamic products contain historical content that subscribers can access. For example, if your product includes a 30-year history of daily closing stock price for U.S. equities, subscribers would get access to that data in addition to the dynamic updates every day.

For these kind of products that contain a historical record of data, a best practice is to publish all historical data in a single revision of the data set. You can use the optional comment for the revision to indicate that this revision is a single upload of all data history from a specific date.

If the single historical revision contains a time series of multiple objects, you might consider labeling your object names to describe the underlying data periodicity. For example, if your single revision of history contains 200 files each with a week of historical data, you can name each file with a date for the week the data history begins.

Suggested Approaches for Updates

You can dynamically update your data sets in a number of ways. Here are three example approaches, all of which create a new revision for each update, but the content of the new revision is different.

- **Use a new revision for each update that contains only the items that have changed since the last revision.** – Your revision size would be smaller because only those items that have changed are updated. This approach is suitable for data sets for which the updates affect only a small subset of the data and subscribers are focused only on the items that have changed.
- **Use a new revision for each update that contains the updated data.** – The new revision contains a full updated file. All items are included in the new revision, including those that have not changed since the last revision. This approach is convenient for subscribers who want to maintain a single up-to-date file for your data. Subscribers export the latest revision's asset or assets to the same destination and override the previous file or files.
- **Use a new revision for each update that contains the full history and updated data.** – The new revision contains the full history of the data, including the latest state of the data and the history of the previous revisions. This approach is more storage-heavy and is suitable for data sets for which subscribers are interested in the latest comprehensive view of the data's history, including any potential past corrections or adjustments. In this approach, each revision is self-sufficient and provides a full view of the data set history with no dependency on previous revisions.

Creating an Offer for AWS Data Exchange Products

To make a product available, you must create an offer in the AWS Data Exchange console. Offers define the terms that subscribers are agreeing to when they subscribe to a product. Each product must have a public offer available to all subscribers. You can also create custom offers for selected subscribers. When you create an offer for your product, you define:

- The data subscription agreement, which defines the terms that a prospective subscriber must agree to before purchasing a subscription for your product.
- Available pricing and duration combinations.
- Whether US sales tax is collected.
- The Terms and Conditions for the refund policy, if any.
- Whether the subscriber must fill out a questionnaire to request a subscription using subscription verification.

You can also create custom offers that you extend to a select AWS account. The custom offer makes it possible for you to set specific terms and pricing for your product. For more information, see Creating Custom Offers (p. 23).

Offer Pricing

When you define the pricing information, you define the total price and duration of the subscription. Durations are 1 to 36 months. You can specify up to 5 different durations in a single offer.

We recommend that you choose durations that you plan to support for the long run. If you discontinue a duration, AWS cancels the subscription renewal for those affected subscribers who opted into an auto-renewal policy.

The only supported currency for pricing is US dollars (USD). You must specify a price for each duration. For example, you can specify different prices for durations of 1 month, 6 months, 12 months, 24 months, and 36 months in a single offer. All options are available to prospective subscribers. They must choose a single price and duration when they subscribe to your offer, and they must agree to your offer terms and pay upfront for the purchase charges.

US Sales and Use Tax

You can enable US sales tax collection for the offer, based on your tax nexus settings. For more information, see U.S. Sales and Use Tax (p. 26).

Data Subscription Agreement

The data subscription agreement (DSA) describes the Terms and Conditions for the data product. As a provider, you control the legal terms and usage rights. These terms are part of each offer you create for your product.

AWS Data Exchange provides a default DSA template. You can download the default DSA template and edit it to add your own Terms and Conditions. Or, you can specify your own custom terms by uploading the DSA of your choice. AWS Data Exchange associates the DSA that you specify for the product's offer without any further modifications.

Refund Policy

As a provider, you control the refund policy for your product's subscribers. Although AWS Data Exchange doesn't require you to offer refunds, you must clearly specify your refund policy in the offer details. We encourage you to provide these details in a clear and concise manner so that subscribers can contact you in case of any questions or requests. AWS can process refunds that you authorize on your behalf, but as the provider, you must authorize the refunds.

For AWS to process authorized refunds, submit a refund approval form to AWS Support through the AWS Marketplace Management Portal. Your refund request is processed and the refund is issued to the subscriber. You can view all refunds that AWS processed on your behalf in the monthly billed revenue report.

Subscription Verification

As a provider, you have the option to enable subscription verification for your data products on AWS Data Exchange. For more information, see Subscription Verification for Providers (p. 25).

Creating Custom Offers

AWS Data Exchange gives providers the option to create custom offers. Currently, the two supported kinds of custom offers are private offers and bring your own subscription (BYOS) offers. For more information about creating these types of offers, see the following topics:

Topics

Create Private Offers

As a data provider, you can provide your data product to a subscriber at terms that are different from the offer terms available to the general public. Private offers allow you to create a custom offer for one or more AWS accounts. A private offer can be different from the public offer in any dimension, including price, duration, data subscription agreement, or refund policy.

As a provider, after you have created a product and an offer that is publicly available to all subscribers, you can then create a private offer and make it available to a group of subscribers of your choosing.

To create a private offer

1. From the left navigation pane of the console, choose **Products**, and then choose the product for which you want to make a private offer.
2. From the **Private offer** tab, choose **Create**.
3. On the **Select Offer Type** page, select **Private offer**, and choose **Next**.
4. Under **Account ID**, enter the 12-digit account number of the account you are creating a private offer for. Because a single private offer can be extended to multiple accounts, you can add more than one account.
5. Under **Description**, provide a short description of the account (for example, the company name of the account).
6. Provide the offer details, including the pricing information, U.S. sales tax and use settings, data subscription agreement, and support information.

   Note
   The **Set expiry date** is the date by which the subscriber must accept the offer. The private offer is no longer available if not accepted by this date.
7. Choose **Next**. After you validate the information is correct, choose **Complete**.

Note
Private offers aren't renewed automatically. You must create another private offer upon expiry.

Create Bring Your Own Subscription (BYOS) Offers

As a data provider, you might already have subscribers for your data products. Bring your own subscription (BYOS) offers allow you to migrate and fulfill existing subscriptions with AWS customers at no additional cost. Before you create a BYOS offer, you need to make the data product publicly available to other AWS customers on AWS Data Exchange.

With BYOS offers, any billing relationship between you and your subscribers continue. BYOS offers are not subject to fulfillment fees. Subscribers receive an AWS Marketplace invoice for the subscription with no charge. After you create a BYOS offer, we review it and contact you if we have any issues or questions.

Because the lifecycle of the subscription begins outside of AWS Data Exchange, the workflow for migrating existing subscription to AWS Data Exchange using BYOS requires collaboration between you and the subscriber.

Important
With BYOS offers, you're migrating a subscription that predates the availability of this product on AWS. AWS might verify your BYOS offer with the existing subscription agreement. If AWS cannot verify your BYOS offer, the offer and entitlements might be revoked without notice.

Here are the steps for creating a BYOS offer on AWS Data Exchange. Before creating or accepting a BYOS offer on AWS Data Exchange, the provider and subscriber should perform the following steps together:

Prerequisites

1. The provider and the subscriber should contact each other about implementing a BYOS AWS Data Exchange solution.
2. The subscriber provides the AWS account ID that they want to use to subscribe to data products on AWS Data Exchange.

If you are the provider, follow these steps to create the BYOS offer. The product must be public before you perform these steps.

To create a BYOS offer

1. Sign in to the AWS Data Exchange console and choose the product for the BYOS offer.
2. Choose **BYOS offers** to open the wizard.
3. Complete the fields in the wizard, including the AWS account ID for the subscriber, and upload the existing DSA or contract you have with the subscriber.
4. Review the offer and acknowledgement before you accept it.
5. Choose **Publish** to create the BYOS offer.
6. Contact the subscriber and tell them the name of the product. This way, the subscriber can search for the offer in the AWS Data Exchange console.

Subscription Verification for Providers

As a provider, you have the option to enable subscription verification for your data product. When enabled, potential subscribers must complete a form about who they are and what they intend to do with the data before they can subscribe. You must review and approve each request from prospective subscribers.

Approving subscription requests to your product can be useful when you have restricted or regulated products, or you have products that you want to limit access to.

The form requires the following information:

- Prospective subscriber's contact details, including contact name, company name, and email address.
- Prospective subscriber's intended use-case.
- Prospective subscriber's AWS account ID.

> **Important**
> The subscriber must enter information in each field, but AWS Data Exchange doesn't review or validate the information. You're solely responsible for reviewing and verifying the information the subscriber provides.

After you receive the subscription request, you have 45 days to approve or reject it. If you don't approve the request in that period of time, the request expires. Potential subscribers can resubmit a rejected request at any time, any number of times.

> **Important**
> The subscriber information you collect through subscription verification must be used in accordance with AWS Marketplace Terms and Conditions.

If you change the product offer terms after a subscriber makes the request, the terms for that subscriber reflect the terms as they were at the time of the request, not the updated terms. Examples of changes to

terms include the price, refund policy, or data subscription agreement. If you changed the product offer terms after the request was submitted, a message is displayed in the approval pane of the AWS Data Exchange console to indicate there is a difference between current terms and the terms in place when the request was made.

The AWS Data Exchange console maintains a history of requests. You control when you delete the subscriber's contact details and personally identifiable information (PII).

You can view all subscription verification requests for all of your products on the **Subscription Verification** tab of the **Products dashboard**.

> **Note**
> Each subscription request is uniquely identified using its ID. The ID is visible to both the provider and the subscriber. You can use the subscription request ID in your communications with the subscriber.

Email Notifications

You will receive an email notification to your AWS account's email address when a request is received, or when its status has changed to cancelled or expired. Although most subscription request status changes result in an email notification, the delivery of these emails is on a best-effort basis.

> **Note**
> You will not receive email notifications for subscription request status changes that you have initiated yourself (for example, when you approve a subscription).

Provider Financials on AWS Marketplace

The following topics cover financial information about providing data through AWS Data Exchange.

AWS Data Exchange is integrated with AWS Marketplace. If you want to register as an AWS Data Exchange provider, you must first register as an AWS Marketplace seller. As an AWS Data Exchange provider, you benefit from AWS Marketplace features, such as Seller Reports and the AWS Marketplace Commerce Analytics Service. For more information, see Seller Reports and Data Feed and AWS Marketplace Enhanced Data Sharing Program in the *AWS Marketplace Seller Guide*.

Payments

AWS disburses payments monthly directly to the bank account associated with the AWS account registered as a seller, minus AWS Marketplace service fees. Payment is disbursed on a rolling monthly basis based on when the account was created, not the beginning of each month. Funds are disbursed to you only after they are collected from the subscriber. For more information, see Disbursement in the *AWS Marketplace Seller Guide*.

U.S. Sales and Use Tax

AWS Marketplace Tax Calculation Service makes it possible to calculate and collect U.S. sales and use tax for existing and new products. Some states are not eligible for Tax Calculation Service because AWS Marketplace is required by law to collect and remit applicable sales tax attributable to taxable sales of your products to subscribers based in these states. To use the service, configure your tax nexus settings for your provider profile, and then assign product tax codes to your products.

To configure your tax nexus settings

- Open the AWS Marketplace Management Portal. On the **Settings** tab, configure the applicable tax nexus settings.

For more information, see Seller Registration Process on the *AWS Marketplace Seller Guide*.

AWS Marketplace Seller Reports

As an AWS Data Exchange provider, you receive reports detailing the subscription activity of your products. There are several reports available to track daily and monthly data. The reports include information about the subscription activity for your offers, payment received from subscribers, and money being disbursed to you. Disbursement doesn't occur until payment is received from the AWS customer. For more information, see Seller Reports in the *AWS Marketplace Seller Guide*.

As an AWS Data Exchange provider, you might be eligible for the AWS Marketplace Enhanced Data Sharing program. For more information, see AWS Marketplace Enhanced Data Sharing Program.

Subscriber Refund Requests

As a provider, you control the refund policy for your products, which you must specify when you create your product. AWS Data Exchange doesn't require you to offer refunds. You must approve all requests for refunds before AWS process them on your behalf.

Submit a refund approval form to AWS Support. They process your request and issue the refund to the subscriber. You can view all refunds that AWS processed on your behalf in the Monthly Billed Revenue Report.

Working with Data Sets

This section covers AWS Data Exchange data sets, their components, properties, and related conceptual information. A data set is a collection of data that can change over time.

- **Data Sets** – A data set is a series of one or more revisions. When you access a data set, you're typically accessing a specific revision in the data set. This structure enables providers to change the data available in data sets over time without having to worry about changes to historical data.
- **Revisions** – A revision is a container for one or more assets. For example, a collection of CSV files or a single CSV file and a dictionary are grouped to create a revision. As new data is available, you create revisions and add assets. For more information, see Revisions (p. 30).
- **Assets** – An asset in AWS Data Exchange is a piece of data that can be stored as an Amazon S3 object. The asset can be a structured data file, an image file, or some other data file. When you upload files to AWS Data Exchange, you create an asset in AWS Data Exchange for each of those files. For more information, see Assets (p. 31).

You can use the AWS Data Exchange console, AWS CLI, your own REST client, or one of the AWS SDKs to create, view, update, or delete data sets. For more information about programmatically managing AWS Data Exchange data sets, see the AWS Data Exchange API Reference.

Owned Data Sets

A data set is owned by the account that created it. Owned data sets can be identified using the origin parameter, which is set to `OWNED`.

Entitled Data Sets

Entitled data sets are a read-only view of owned data sets. Entitled data sets are created at time of product publishing and are made available to subscribers who have an active subscription to the product. Entitled data sets can be identified using the origin parameter, which is set to `ENTITLED`.

As a data subscriber, you can view and interact with your entitled data sets using the AWS Data Exchange APIs, or in the Console.

As a data provider, you also have access to the entitled data set view that your subscribers see. You can do so using the AWS Data Exchange APIs, or the Console, by choosing the data set name in the product page.

AWS Regions and Data Sets

Your data sets can be in any supported AWS Region, but all data sets in a single product must be in the same AWS Region.

Data Sets Published to Multiple Products

As a provider, you can add the same data set to multiple products, and choose the revisions that are available in each product. For example, if you offer data that is updated regularly, you can provide it in two products: one updated weekly and the other daily.

If the same data set is published into multiple products, a distinct entitled data set is created for each product, each with a distinct data set ID.

Tags

You can add tags to your owned data sets and their revisions. When you use tagging, you can also use tag-based access control in IAM policies to control access to these data sets and revisions.

Entitled data sets can't be tagged. Tags of owned data sets and their revisions are not propagated to their corresponding entitled versions. Specifically, subscribers, who have read-only access to entitled data sets and revisions, won't see the tags of the original owned data set.

Note
Currently, assets and jobs don't support tagging.

Data Set Structure

Data sets have the following parameters:

- `Name` – The name of the data set. This value can be up to 256 characters long.
- `Description` – A description for the data set. This value can be up to 16,348 characters long.
- `Asset type` – Defines the type of assets the data set contains. Currently, the only supported asset type is snapshots of Amazon S3 objects.
- `Origin` – A property that defines the data set as `Owned` by the account (for providers) or `Entitled` to the account (for subscribers).
- `Data set ID` – An ID that uniquely identifies the data set. Data set IDs are generated at data set creation. Entitled data sets have a different ID than the original owned data set.
- `Amazon Resource Name (ARN)` – A unique identifier for an AWS resource.
- `Created and updated dates` – Timestamps for the creation and last update of the data set.

Note
As a provider, you can change some properties for owned data sets, like the name or description. Updating properties in an owned data set won't update the properties in the corresponding entitled data set.

Example Data Set Resource

```
{
    "Origin": "OWNED",
    "AssetType": "S3_SNAPSHOT",
    "Name": "MyDataSetName",
    "CreatedAt": "2019-09-09T19:31:49.704Z",
    "UpdatedAt": "2019-09-09T19:31:49.704Z",
    "Id": "fEXAMPLE1fd9a5c8b0d2e6fEXAMPLEe1",
    "Arn": "arn:aws:dataexchange:us-east-2:123456789109:data-
sets/fEXAMPLE1fd9a5c8b0d2e6fEXAMPLEe1",
```

```
    "Description": "This is my data set's description that describes the contents of the
 data set."
}
```

Data Set Best Practices

As a provider, when you create and update data sets, keep the following best practices in mind:

- The name of the data set is visible in the product details in the catalog. We recommend that you choose a concise, descriptive name so customers easily understand the content of the data set.
- The description is visible to subscribers who have an active subscription to the product. We recommend that you include coverage information and the features and benefits of the data set.

Revisions

Data sets can be updated over time. When you want to add or change a file in a data set, you create a revision. You can create and add revisions programmatically or through the console.

When they're created, revisions are not published to products, and therefore are not available to subscribers. To publish a revision to a data set in a product, the revision must first be *finalized*. Finalizing a revision tells AWS Data Exchange that your changes to the assets in the revision are complete. After the revision is in this finalized (read-only) state, you can publish it to your products.

You can use the AWS Data Exchange console or the AWS Marketplace Catalog API to publish finalized revisions. If you choose the API, use the StartChangeSet AWS Marketplace Catalog API action. Revisions are uniquely identified by their ARN.

Keep the following in mind:

- A revision must be marked as finalized before it can be published to a product.
- To be finalized, a revision must contain at least one asset.
- It is your responsibility to ensure that the assets are correct before you finalize your revision.
- A finalized revision is staged for publishing. No changes can be made to it in the finalized state.
- Before a finalized revision is published, it can be unfinalized, allowing you to make changes. After you make your changes, you can finalize it again for publishing.
- A finalized revision published to at least one product cannot be unfinalized or changed in any way.

> **Important**
> Any revision published to a product is immutable and can't be edited, changed, or deleted. If you need to remove published content for compliance reasons, contact AWS Support.

Revision Structure

Revisions have the following parameters:

- `Data set ID` – The ID of the data set that contains this revision.
- `Comment` – A comment about the revision. This field can be 128 characters long.
- `Finalized state` – Either true or false. Used to indicate whether the revision is finalized.
- `ID` – The unique identifier for the revision generated when it's created.
- `Amazon Resource Name (ARN)` – A unique identifier for an AWS resource.

- `Created and updated dates` – Timestamps for the creation and last update of the revision. Entitled revisions are created at the time of publishing.

Example Revision Resource

```
        {
            "UpdatedAt": "2019-10-11T14:13:31.749Z",
            "DataSetId": "1EXAMPLE404460dc9b005a0d9EXAMPLE2f",
            "Comment": "initial data revision",
            "Finalized": true,
            "Id": "e5EXAMPLE224f879066f9999EXAMPLE42",
            "Arn": "arn:aws:dataexchange:us-east-1:123456789012:data-
sets/1EXAMPLE404460dc9b005a0d9EXAMPLE2f/revisions/e5EXAMPLE224f879066f9999EXAMPLE42",
            "CreatedAt": "2019-10-11T14:11:58.064Z"
        }
```

Assets

Assets are the *data* in AWS Data Exchange. Each asset is a snapshot of an Amazon S3 object, with a maximum size of 10 GB. You can use the console, or programmatically through the AWS CLI, your own REST application, or one of the AWS SDKs to create or copy assets through jobs.

A data set owner can both import and export, but someone with an entitlement to a data set can only export.

Asset Structure

Assets have the following parameters:

- `Data set ID` – The ID of the data set that contains this asset.
- `Revision ID` – The ID of the revision that contains this asset.
- `ID` – A unique ID generated when the asset is created.
- `Amazon Resource Name (ARN)>` – A uniquely identifier for an AWS resource.
- `Created and updated dates` – Timestamps for the creation and last update of the asset.
- `Asset details` – Information about the asset, including its size.
- `Asset Type` – Currently, the only type of asset available is a snapshot of an Amazon S3 object.

Example Asset Resource

```
{
    "Name": "automation/cloudformation.yaml",
    "Arn": "arn:aws:dataexchange:us-east-1::data-sets/29EXAMPLE24b82c6858af3cEXAMPLEcf/
revisions/bbEXAMPLE74c02f4745c660EXAMPLE20/assets/baEXAMPLE660c9fe7267966EXAMPLEf5",
    "Id": "baEXAMPLE660c9fe7267966EXAMPLEf5",
    "CreatedAt": "2019-10-17T21:31:29.833Z",
    "UpdatedAt": "2019-10-17T21:31:29.833Z",
    "AssetType": "S3_SNAPSHOT",
    "RevisionId": "bbEXAMPLE74c02f4745c660EXAMPLE20",
    "DataSetId": "29EXAMPLE24b82c6858af3cEXAMPLEcf",
    "AssetDetails": {
        "S3SnapshotAsset": {
            "Size": 9423
        }
```

```
    }
}
```

Jobs

AWS Data Exchange jobs are asynchronous import or export operations used to create or copy assets. A data set owner can import and export, but someone with an entitlement to a data set can only export. You can use the console, AWS CLI, your own REST application, or one of the AWS SDKs to create or copy assets through jobs.

Jobs are deleted 90 days after they are created.

Topics

Job Properties

Jobs have the following properties:

- **Job ID** – An ID generated when the job is created that uniquely identifies the job.
- **Job type** – The following job types are supported: import from Amazon S3, import from signed URL, export to Amazon S3, export to signed URL.
- **Amazon Resource Name (ARN)**> – A unique identifier for AWS resources.
- **Job state** – The job state can be: `WAITING`, `IN_PROGRESS`, `COMPLETED`, `CANCELLED`, `ERROR`, or `TIMED_OUT`. When a job is created, it's put in the `WAITING` state until the job is started.
- **Job details** – Details of the operation to be performed by the job, such as export destination details or import source details.

Example Job Resource

```
{
    "Arn": "arn:aws:dataexchange:us-
east-1:123456789012:jobs/6cEXAMPLE818f7c7a23b3d0EXAMPLE1c",
    "Id": "6cEXAMPLE818f7c7a23b3d0EXAMPLE1c",
    "State": "COMPLETED",
    "Type": "IMPORT_ASSETS_FROM_S3",
    "CreatedAt": "2019-10-11T14:12:24.640Z",
    "UpdatedAt": "2019-10-11T14:13:00.804Z",
    "Details": {
        "ImportAssetsFromS3": {
            "AssetSources": [
                {
                    "Bucket": "awsexamplebucket",
                    "Key": "MyKey"
                }
            ],
            "DataSetId": "14EXAMPLE4460dc9b005a0dEXAMPLE2f",
            "RevisionId": "e5EXAMPLE224f879066f999EXAMPLE42"
        }
    }
```

```
}
```

AWS Regions and Jobs

If you import or export an asset to or from an Amazon S3 bucket that is in an AWS Region different from the data set's, your AWS account is charged for the data transfer costs according to Amazon S3 data transfer pricing policies.

Importing Assets

There are two ways you can import assets to a revision:

- From an Amazon S3 bucket that you have permissions to access.
- By using a signed URL.

Importing Assets from an Amazon S3 Bucket

When you import from an Amazon S3 bucket, you must create and start a job of type `IMPORT_ASSETS_FROM_S3`. Provide the details of the import destinations (including the asset ID, revision ID, and data set ID) and the asset sources (Amazon S3). The newly created assets have a name property equal to the original S3 object's key. You can update the assets' name property after they are created. You can import up to 100 assets in a single job.

When importing assets from Amazon S3 to AWS Data Exchange, the IAM permissions you're using must include the ability to write to the AWS Data Exchange service Amazon S3 buckets and to read from the Amazon S3 bucket where your assets are stored. You can import from any Amazon S3 bucket you have permission to access, regardless of ownership. For more information, see Additional Amazon S3 Permissions (p. 43).

Importing Assets from a Signed URL

You can use signed URLs to import assets that are not stored in Amazon S3. Create a job of type `IMPORT_ASSET_FROM_SIGNED_URL`, provide the 24-byte MD5 hash of the asset, and the asset name. The job's details include a signed URL that you can use to import your file. The signed URL expires one hour after it's created.

Exporting Assets

There are two ways you can export assets from a published revision of a product:

- To an Amazon S3 bucket that you have permissions to access.
- By using a signed URL.

Exporting Assets to an Amazon S3 Bucket

When you export to an Amazon S3 bucket, you must create and start a job of type `EXPORT_ASSETS_TO_S3`. Provide details of the assets you would like to export and the target destination. By default, the assets are exported to an S3 object using the original asset name as an object key. You can export up to 100 assets in a single job.

AWS Data Exchange supports configurable encryption parameters when exporting data sets to Amazon S3. In your export job details, you can specify the Amazon S3 server-side encryption configuration you want to apply to the exported objects. You can choose to use server-side encryption with Amazon S3-

Managed Keys (SSE-S3) or server-side encryption with Customer Master Keys (CMKs) stored in AWS Key Management Service (SSE-KMS). For more information, see Protecting data using server-side encryption in the *Amazon Simple Storage Service Developer Guide*.

Note
When exporting assets to Amazon S3, the IAM permissions you're using must include the ability to read from the AWS Data Exchange service Amazon S3 buckets and to write to the Amazon S3 bucket where your assets are stored. You can export to any Amazon S3 bucket you have permission to access, regardless of ownership. For more information, see Additional Amazon S3 Permissions (p. 43).

Exporting Assets to a Signed URL

You can use signed URLs to export assets to destinations other than S3 buckets. Create and start a job of type `EXPORT_ASSET_TO_SIGNED_URL` and provide the source details. The job's details include a signed URL that you can use to export your file. The signed URL has an expiry time of 1 minute.

AWS Data Exchange Limits

The following tables lists limits for AWS Data Exchange for an AWS account.

Service Limits

Resource, Descriptor, or Operation	Default Limit	Description
Products per account	50	The maximum number of products per account.
Data sets per account	3,000	The maximum number of data sets per account.
Products per data set	100	The maximum number of products that can contain a given data set.
Concurrent export jobs (to S3 or a signed URL)	10	The maximum concurrent number of running export jobs with the `IN_PROGRESS` state.
Concurrent import jobs (from S3 or a signed URL)	10	The maximum concurrent number of running import jobs with the `IN_PROGRESS` state.
Private offers per account	10	The maximum number of private offers that a single account can create.
BYOS offers per account	10	The maximum number of BYOS offers that a single account can create.
Revisions per `addRevisions` change set	5	The maximum number of revisions that can be published to a product in a single AWS Marketplace Catalog API `ChangeSet` of type `addRevisions`.
Number of assets that can be imported or exported to/from Amazon S3 in a single job	100	A single job can import or export up to 100 assets to or from Amazon S3.
Number of assets per single revision	10,000	A single revision can contain up to 10,000 assets.
Number of revisions per single data set	10,000	A single data set can contain up to 10,000 revisions.
Number of data sets per single product	25	A product can have up to 25 data sets.

Resource, Descriptor, or Operation	Default Limit	Description
Asset size in GB	10 GB	The maximum size, in GB, of a single asset.

Limits on Resource Fields

Resources	Field	Maximum Length
Data set	Name	256 characters
Data set	Description	16,384 characters
Revision	Comment	128 characters
Product	Name	72 characters
Product	Short description	500 characters
Product	Long description	30,000 characters
Product	Logo	100 KB
Offer	DSA	10 MB
Offer	Refund policy	200 characters
Subscription request	company name	40 characters
Subscription request	name	40 characters
Subscription request	email address	100 characters
Subscription request	intended use-case	500 characters

Endpoints and AWS Regions

The following AWS Regions are endpoints are supported for AWS Data Exchange:

- US East (N. Virginia) – `dataexchange.us-east-1.amazonaws.com`
- US East (Ohio) – `dataexchange.us-east-2.amazonaws.com`
- US West (N. California) – `dataexchange.us-west-1.amazonaws.com`
- US West (Oregon) – `dataexchange.us-west-2.amazonaws.com`
- Asia Pacific (Tokyo) – `dataexchange.ap-northeast-1.amazonaws.com`
- Asia Pacific (Seoul) – `dataexchange.ap-northeast-2.amazonaws.com`
- Asia Pacific (Singapore) – `dataexchange.ap-southeast-1.amazonaws.com`
- Asia Pacific (Sydney) – `dataexchange.ap-southeast-2.amazonaws.com`
- Europe (Frankfurt) – `dataexchange.eu-central-1.amazonaws.com`
- Europe (Ireland) – `dataexchange.eu-west-1.amazonaws.com`
- Europe (London) – `dataexchange.eu-west-2.amazonaws.com`

Security

Cloud security at AWS is the highest priority. As an AWS customer, you benefit from multiple data centers and network architecture that is built to meet the requirements of the most security-sensitive organizations.

Security is a shared responsibility between AWS and you. The shared responsibility model describes this as security of the cloud and security in the cloud:

- **Security of the cloud** – AWS is responsible for protecting the infrastructure that runs AWS services in the AWS Cloud. AWS also provides you with services that you can use securely. The effectiveness of our security is regularly tested and verified by third-party auditors as part of AWS compliance programs. To learn about the compliance programs that apply to AWS Data Exchange, see AWS Services in Scope by Compliance Program.
- **Security in the cloud** – Your responsibility is determined by the AWS services that you use. You are also responsible for other factors, including the sensitivity of your data, your organization's requirements, and applicable laws and regulations.

This documentation helps you understand how to apply the shared responsibility model when you use AWS Data Exchange. The following topics show you how to configure AWS Data Exchange to meet your security and compliance objectives. You also learn how to use other AWS services that help you monitor and secure your AWS Data Exchange resources.

Data Protection in AWS Data Exchange

AWS Data Exchange conforms to the AWS shared responsibility model, which includes regulations and guidelines for data protection. AWS is responsible for protecting the global infrastructure that runs all AWS services. AWS maintains control over data hosted on this infrastructure, including the security configuration controls for handling customer content and personal data. AWS customers and APN partners, acting either as data controllers or data processors, are responsible for any personal data that they put in the AWS Cloud.

For data protection purposes, we recommend that you protect AWS account credentials and set up individual user accounts with AWS Identity and Access Management (IAM), so that each user is given only the permissions required to fulfill their job duties. We also recommend that you secure your data in the following ways:

- Use multi-factor authentication (MFA) with each account.
- Use SSL/TLS to communicate with AWS resources.
- Set up API and user activity logging with CloudTrail.
- Use AWS encryption solutions, along with all default security controls in AWS services.
- Use advanced managed security services such as Amazon Macie, which assists in discovering and securing personal data that is stored in Amazon S3.

We strongly recommend that you never put sensitive identifying information, such as your customers' account numbers, into free-form fields such as a **Name** field. This includes when you work with AWS Data Exchange or other AWS services using the console, API, AWS CLI, or AWS SDKs. Any data that you enter into AWS Data Exchange or other services might get picked up for inclusion in diagnostic logs. When you provide a URL to an external server, don't include credentials information in the URL to validate your request to that server.

For more information about data protection, see the AWS Shared Responsibility Model and GDPR blog post on the *AWS Security Blog*.

AWS Data Exchange provides the following options that you can use to help secure the content that exists in your data sets:

Topics

Encryption at Rest

AWS Data Exchange always encrypts all data products stored in the service at rest without requiring any additional configuration. This encryption is automatic when you use AWS Data Exchange.

Encryption in Transit

AWS Data Exchange uses Transport Layer Security (TLS) and client-side encryption for encryption in transit. Communication with AWS Data Exchange is always done over HTTPS so your data is always encrypted in transit. This encryption is configured by default when you use AWS Data Exchange.

Restrict Access to Content

As a best practice, you should restrict access to the appropriate subset of users. With AWS Data Exchange, you can do this by ensuring that IAM users, groups, and roles who use your AWS account have the right permissions. For more information about roles and policies for IAM entities, see *IAM User Guide*.

Identity and Access Management in AWS Data Exchange

To perform any operation in AWS Data Exchange, such as creating an import job using an AWS SDK, or subscribing to a product in the AWS Data Exchange console, AWS Identity and Access Management (IAM) requires that you to authenticate that you're an approved AWS user. For example, if you're using the AWS Data Exchange console, you authenticate your identity by providing your AWS user name and a password.

After you authenticate your identity, IAM controls your access to AWS with a defined set of permissions on a set of operations and resources. If you are an account administrator, you can use IAM to control the access of other IAM users to the resources that are associated with your account.

Topics

Authentication

You can access AWS with any of the following types of identities:

- **AWS account root user** – When you first create an AWS account, you begin with an identity that has complete access to all AWS services and resources in the account. This identity is called the AWS account root user and is accessed by signing in with the email address and password that you used to create the account. We strongly recommend that you do not use the root user for your everyday tasks, even the administrative ones. Instead, adhere to the best practice of using the root user only to create your first IAM user. Then securely lock away the root user credentials and use them to perform only a few account and service management tasks.
- **IAM user** – An IAM user is an identity in your AWS account that has specific custom permissions. You can use an IAM user name and password to sign in to secure AWS webpages like the AWS Management Console, AWS Discussion Forums, or the AWS Support Center.

  In addition to a user name and password, you can also generate access keys for each user. You can use these keys when you access AWS services programmatically, either through one of the several SDKs or by using the AWS Command Line Interface (CLI). The SDK and CLI tools use the access keys to cryptographically sign your request. If you don't use AWS tools, you must sign the request yourself. AWS Data Exchange supports Signature Version 4, a protocol for authenticating inbound API requests. For more information about authenticating requests, see Signature Version 4 Signing Process in the *AWS General Reference*.
- **IAM role** – An IAM role is an IAM identity that you can create in your account that has specific permissions. An IAM role is similar to an IAM user in that it is an AWS identity with permissions policies that determine what the identity can and cannot do in AWS. However, instead of being uniquely associated with one person, a role is intended to be assumable by anyone who needs it. Also, a role does not have standard long-term credentials, such as a password or access keys, associated with it. Instead, when you assume a role, it provides you with temporary security credentials for your role session. IAM roles with temporary credentials are useful in the following situations:
  - **Federated user access** – Instead of creating an IAM user, you can use existing identities from AWS Directory Service, your enterprise user directory, or a web identity provider. These are known as *federated users*. AWS assigns a role to a federated user when access is requested through an identity provider. For more information about federated users, see Federated Users and Roles in the *IAM User Guide*.
  - **AWS service access** – A service role is an IAM role that a service assumes to perform actions in your account on your behalf. When you set up some AWS service environments, you must define a role for the service to assume. This service role must include all the permissions that are required for the service to access the AWS resources that it needs. Service roles vary from service to service, but many allow you to choose your permissions as long as you meet the documented requirements for that service. Service roles provide access only within your account and cannot be used to grant access to services in other accounts. You can create, modify, and delete a service role from within IAM. For example, you can create a role that allows Amazon Redshift to access an Amazon S3 bucket on your behalf and then load data from that bucket into an Amazon Redshift cluster. For more information, see Creating a Role to Delegate Permissions to an AWS Service in the *IAM User Guide*.
  - **Applications running on Amazon EC2** – You can use an IAM role to manage temporary credentials for applications that are running on an Amazon EC2 instance and making AWS CLI or AWS API requests. This is preferable to storing access keys in the Amazon EC2 instance. To assign an AWS role to an Amazon EC2 instance and make it available to all of its applications, you create an instance profile that is attached to the instance. An instance profile contains the role and enables programs that are running on the Amazon EC2 instance to get temporary credentials. For more information, see Using an IAM Role to Grant Permissions to Applications Running on Amazon EC2 Instances in the *IAM User Guide*.

Access Control

To create, update, delete, or list AWS Data Exchange resources, you need permissions to perform the operation and to access the corresponding resources. To perform the operation programmatically, you also need valid access keys.

Overview of Managing Access Permissions to Your AWS Data Exchange Resources

Every AWS resource is owned by an AWS account, and permissions to create or access a resource are governed by permissions policies. An account administrator can attach permissions policies to IAM identities (that is, users, groups, and roles). Some services (such as AWS Lambda) also support attaching permissions policies to resources.

> **Note**
> An *account administrator* (or administrator) is a user with administrator privileges. For more information, see IAM Best Practices in the *IAM User Guide*.

When granting permissions, you decide who is getting the permissions, the resources they get permissions for, and the specific actions that you want to allow on those resources.

Topics

AWS Data Exchange Resources and Operations

In AWS Data Exchange, there are two different kinds of primary resources with different control planes:

- The primary resources for AWS Data Exchange are *data sets* and *jobs*. AWS Data Exchange also supports *revisions* and *assets*.
- To facilitate transactions between providers and subscribers, AWS Data Exchange also uses AWS Marketplace concepts and resources, including products, offers, and subscriptions. You can use the AWS Marketplace Catalog API or the AWS Data Exchange console to manage your products, offers, subscription requests, and subscriptions.

Understanding Resource Ownership

The AWS account owns the resources that are created in the account, regardless of who created the resources. Specifically, the resource owner is the AWS account of the principal entity (that is, the AWS account root user, an IAM user, or an IAM role) that authenticates the resource creation request. The following examples illustrate how this works:

Resource Ownership

Any IAM entity in an AWS account with the right permissions can create AWS Data Exchange data sets. When an IAM entity creates a data set, their AWS account owns the data set. Published data products can contain data sets that are owned only by the AWS account that created them.

To subscribe to an AWS Data Exchange product, the IAM entity will need permissions use AWS Data Exchange, as well as the `aws-marketplace:subscribe` IAM permission for AWS Marketplace (assuming they pass any related subscription verifications). As a subscriber, your account has read access to entitled data sets, however it does not own the entitled data sets. Any entitled data sets that are exported to Amazon S3 are owned by the subscriber's AWS account.

Managing Access to Resources

This section discusses using IAM in the context of AWS Data Exchange. It doesn't provide detailed information about the IAM service. For complete IAM documentation, see What Is IAM? in the *IAM User*

Guide. For information about IAM policy syntax and descriptions, see AWS IAM Policy Reference in the *IAM User Guide*.

A *permissions policy* describes who has access to what. The following section explains the options for creating permissions policies.

Policies attached to an IAM identity are referred to as *identity-based* policies (IAM polices). Policies attached to a resource are referred to as *resource-based* policies. AWS Data Exchange supports only identity-based policies (IAM policies).

Topics

Identity-Based Policies (IAM Policies)

You can attach policies to IAM identities. For example, you can do the following:

- **Attach a permissions policy to a user or a group in your account** – To grant a user permissions to create an AWS Data Exchange resource, like a revision, you can attach a permissions policy to a user or group that the user belongs to.
- **Attach a permissions policy to a role (grant cross-account permissions)** – You can attach an identity-based permissions policy to an IAM role to grant cross-account permissions. For example, the administrator in Account A can create a role to grant cross-account permissions to another AWS account (for example, Account B) or an AWS service as follows:
  1. Account A administrator creates an IAM role and attaches a permissions policy to the role that grants permissions on resources in Account A.
  2. Account A administrator attaches a trust policy to the role identifying Account B as the principal who can assume the role.
  3. Account B administrator can then delegate permissions to assume the role to any users in Account B. Doing this allows users in Account B to create or access resources in Account A. The principal in the trust policy can also be an AWS service principal, if you want to grant an AWS service permissions to assume the role.

  For more information about using IAM to delegate permissions, see Access Management in the *IAM User Guide*.

AWS Data Exchange provides four managed policies:

- `AWSDataExchangeFullAccess`
- `AWSDataExchangeSubscriberFullAccess`
- `AWSDataExchangeProviderFullAccess`
- `AWSDataExchangeReadOnly`

The following is an example managed policy, `AWSDataExchangeProviderFullAccess`. It grants an entity full access to AWS Data Exchange and the permissions required for other related services to perform all provider-related actions on AWS Data Exchange:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
```

```
            "dataexchange: CreateDataSet",
            "dataexchange: CreateRevision",
            "dataexchange: Get*",
            "dataexchange: Update*",
            "dataexchange: List*",
            "dataexchange: Delete*"
        ],
        "Resource": "*"
    },
    {
        "Effect": "Allow",
        "Action": [
            "dataexchange: CreateJob",
            "dataexchange: StartJob",
            "dataexchange: CancelJob"
        ],
        "Resource": "*",
        "Condition": {
            "dataexchange: JobType": [
                "IMPORT_ASSETS_FROM_S3",
                "IMPORT_ASSET_FROM_SIGNED_URL",
                "EXPORT_ASSETS_TO_S3",
                "EXPORT_ASSET_TO_SIGNED_URL"
            ]
        }
    },
    {
        "Effect": "Allow",
        "Action": [
            "s3: GetObject"
        ],
        "Resource": "*",
        "Condition": {
            "StringEqualsIgnoreCase": {
                "s3: ExistingObjectTag/aws-data-exchange": "true"
            }
        }
    },
    {
        "Action": "s3: GetObject",
        "Effect": "Allow",
        "Resource": "arn: aws: s3: : : *aws-data-exchange*"
    },
    {
        "Action": [
            "s3: PutObject",
            "s3: PutObjectAcl"
        ],
        "Effect": "Allow",
        "Resource": "arn: aws: s3: : : *aws-data-exchange*"
    },
    {
        "Effect": "Allow",
        "Action": [
            "s3: GetBucketLocation"
        ],
        "Resource": "*"
    },
    {
        "Effect": "Allow",
        "Action": [
            "aws-marketplace: ListEntities",
            "aws-marketplace: DescribeEntity",
            "aws-marketplace: StartChangeSet",
            "aws-marketplace: ListChangeSets",
            "aws-marketplace: DescribeChangeSet",
```

```
                "aws-marketplace: CancelChangeSet",
                "aws-marketplace: ListAgreementApprovalRequests",
                "aws-marketplace: GetAgreementApprovalRequest",
                "aws-marketplace: AcceptAgreementApprovalRequest",
                "aws-marketplace: RejectAgreementApprovalRequest",
                "aws-marketplace: UpdateAgreementApprovalRequest"
            ],
            "Resource": "*"
        }
    ]
}
```

Additional Amazon S3 Permissions

When importing assets from Amazon S3 to AWS Data Exchange you need permissions to write to the AWS Data Exchange service Amazon S3 buckets. Similarly, when exporting assets from AWS Data Exchange to Amazon S3, you need permissions to read from the AWS Data Exchange service Amazon S3 buckets. You can scope these permissions to buckets that contain `aws-data-exchange` in their name.

For example, the `AWSDataExchangeProviderFullAccess` managed policy shown previously includes these permissions:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject"
            ],
            "Resource": "*",
            "Condition": {
                "StringEqualsIgnoreCase": {
                    "s3:ExistingObjectTag/aws-data-exchange": "true"
                }
            }
        },
        {
            "Action": "s3:GetObject",
            "Effect": "Allow",
            "Resource": "arn:aws:s3:::*aws-data-exchange*"
        },
        {
            "Action": [
                "s3:PutObject",
                "s3:PutObjectAcl"
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:s3:::*aws-data-exchange*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetBucketLocation"
            ],
            "Resource": "*"
        }
    ]
}
```

These permissions allow providers to import and export with AWS Data Exchange. The policy includes the following permissions:

- **s3:putObject and s3:putObjectAcl** – These permissions are restricted only to Amazon S3 buckets that contain `aws-data-exchange` in their name. These permissions allows providers to write to AWS Data Exchange service buckets when importing from Amazon S3.
- **s3:getBucketLocation** – This permission on the Amazon S3 buckets from which assets are imported allows providers to optimize import speed. This permission is also needed to use the AWS Data Exchange console when importing.
- **s3:getObject** – This permission is restricted to:
  - Amazon S3 objects that are tagged with `"aws-data-exchange":"true"`. This permission allows providers to read the Amazon S3 objects they are importing.
  - Amazon S3 buckets that contain `aws-data-exchange` in their name. This permission allows providers to write to AWS Data Exchange service buckets when exporting out of AWS Data Exchange to Amazon S3.

For more information about users, groups, roles, and permissions, see Identities (Users, Groups, and Roles) in the *IAM User Guide*.

Resource-Based Policies

Other services, such as Amazon S3, also support resource-based permissions policies. For example, you can attach a policy to an S3 bucket to manage access permissions to that bucket.

Specifying Policy Elements: Actions, Effects, and Principals

To use AWS Data Exchange, you must be an IAM user with the appropriate permissions defined in a IAM policy.

The following are the most basic policy elements:

- **Resource** – In a policy, you use an Amazon Resource Name (ARN) to identify the resource to which the policy applies. All AWS Data Exchange APIs support resource level permissions (RLP), but AWS Marketplace actions don't support RLP. For more information, see AWS Data Exchange Resources and Operations (p. 40).
- **Action** – You use action keywords to identify resource operations that you want to allow or deny.
- **Effect** – You specify the effect (allow or deny) when the user requests the specific action. If you don't explicitly grant access to (allow) a resource, access is implicitly denied. You can also explicitly deny access to a resource, which you might do to make sure that a user cannot access it, even if a different policy grants access.
- **Principal** – In identity-based policies (IAM policies), the user that the policy is attached to is the implicit principal. For resource-based policies, you specify the user, account, service, or other entity that you want to receive permissions (applies to resource-based policies only). AWS Data Exchange doesn't support resource-based policies.

For more information about IAM policy syntax and descriptions, see AWS IAM Policy Reference in the *IAM User Guide*.

Specifying Conditions in a Policy

When you grant permissions, you can use the IAM policy language to specify the conditions when a policy should take effect. With AWS Data Exchange, the `CreateJob`, `StartJob`, `GetJob`, `CancelJob` APIs support conditional permissions. You can provide permissions at the `JobType` level.

AWS Data Exchange Condition Key Reference

Condition Key	Description	Type
"dataexchange:JobType":"IMPORT_ASSETS_FROM_S3"	Scopes permissions to jobs that import assets from Amazon S3.	String
"dataexchange:JobType":"IMPORT_ASSETS_FROM_SIGNED_URL"	Scopes permissions to jobs that import assets from a signed URL.	String
"dataexchange:JobType":"EXPORT_ASSETS_TO_S3"	Scopes permissions to jobs that export assets to Amazon S3.	String
"dataexchange:JobType":"EXPORT_ASSETS_TO_SIGNED_URL"	Scopes permissions to jobs that export assets to a signed URL.	String

For more information about specifying conditions in a policy language, see Condition in the *IAM User Guide*.

To express conditions, you use predefined condition keys. AWS Data Exchange has the `JobType` condition for APIs. However, there are AWS-wide condition keys that you can use, as appropriate. For a complete list of AWS-wide keys, see Available Keys for Conditions in the *IAM User Guide*.

AWS Data Exchange API Permissions: Actions and Resources Reference

Use the following table as a reference when you are setting up Access Control (p. 39) and writing a permissions policy that you can attach to an IAM identity (identity-based policies). The table lists each AWS Data Exchange API operation, the actions for which you can grant permissions to perform the action, and the AWS resource for which you can grant the permissions. You specify the actions in the policy's `Action` field. You specify the resource value in the policy's `Resource` field.

> **Note**
> To specify an action, use the `dataexchange:` prefix followed by the API operation name (for example, `dataexchange:CreateDataSet`).

AWS Data Exchange API and Required Permissions for Actions

AWS Data Exchange API Operations	Required Permissions (API Actions)	Resources	Conditions
CreateDataSet	`dataexchange:CreateDataSet`	N/A	`aws:TagKeys` `aws:RequestTag`
GetDataSet	`dataexchange:GetDataSet`	DataSet	`aws:RequestTag`
UpdateDataSet	`dataexchange:UpdateDataSet`	DataSet	`aws:RequestTag`
DeleteDataSet	`dataexchange:DeleteDataSet`	DataSet	`aws:RequestTag`
ListDataSet	`dataexchange:ListDataSet`	N/A	N/A
CreateRevision	`dataexchange:CreateRevision`	DataSet	`aws:TagKeys`

AWS Data Exchange API Operations	Required Permissions (API Actions)	Resources	Conditions
			aws:RequestTag
GetRevision	dataexchange:GetRevision	Revision	aws:RequestTag
DeleteRevision	dataexchange:DeleteRevision	Revision	aws:RequestTag
ListDataSetRevisions	dataexchange:ListDataSetRevisions	DataSet	aws:RequestTag
ListRevisionAssets	dataexchange:ListRevisionAssets	Revision	aws:RequestTag
CreateJob	dataexchange:CreateJob	N/A	dataexchange:JobType
GetJob	dataexchange:GetJob	Job	dataexchange:JobType
StartJob**	dataexchange:StartJob	Job	dataexchange:JobType
CancelJob	dataexchange:CancelJob	Job	dataexchange:JobType
ListJob	dataexchange:ListJobs	N/A	N/A
ListTagsForResource	dataexchange:ListTagsForResource	Revision	aws:RequestTag
TagResource	dataexchange:TagResource	Revision	aws:TagKeys aws:RequestTag
UnTagResource	dataexchange:UnTagResource	Revision	aws:TagKeys aws:RequestTag
UpdateRevision	dataexchange:UpdateRevision	Revision	aws:RequestTag
DeleteAsset	dataexchange:DeleteAsset	Asset	N/A
GetAsset	dataexchange:GetAsset	Asset	N/A
UpdateAsset	dataexchange:UpdateAsset	Asset	N/A

** Additional IAM permissions might be needed depending on the type of the job you are starting. See the table below for the AWS Data Exchange job types and associated additional IAM permissions. For more information on jobs, see Jobs (p. 32).

AWS Data Exchange Job Type Permissions for `StartJob`

Job Type	Additional IAM Permissions Needed
IMPORT_ASSETS_FROM_S3	dataexchange:CreateAsset
IMPORT_ASSETS_FROM_SIGNED_URL	dataexchange:CreateAsset
EXPORT_ASSETS_TO_S3	dataexchange:GetAsset
EXPORT_ASSETS_TO_SIGNED_URL	dataexchange:GetAsset

You can scope data set actions to the revision or asset level through the use of wildcards, as in the following example:

```
arn:aws:dataexchange:us-east-1:123456789012:data-sets/99EXAMPLE23c7c272897cf1EXAMPLE7a/
revisions/*/assets/*
```

Some AWS Data Exchange actions can only be performed on the AWS Data Exchange console. These actions are integrated with AWS Marketplace functionality and require the following AWS Marketplace permissions:

AWS Data Exchange console-only actions for subscribers

Console action	IAM permission
Subscribe to a product	`aws-marketplace:Subscribe`
Send subscription verification request	`aws-marketplace:Subscribe`
Enable subscription auto-renew	`aws-marketplace:Subscribe`
Disable subscription auto-renew	`aws-marketplace:Unsubscribe`
List active subscriptions	`aws-marketplace:ViewSubscriptions`
View subscription	`aws-marketplace:ViewSubscriptions`
List subscription verification requests	`aws-marketplace:ListAgreementRequests`
View subscription verification request	`aws-marketplace:GetAgreementRequest`
Cancel subscription verification request	`aws-marketplace:CancelAgreementRequest`

AWS Data Exchange console-only actions for providers

Console action	IAM permission
Publish product	`aws-marketplace:StartChangeSet` `aws-marketplace:DescribeChangeSet`
Unpublish product	`aws-marketplace:StartChangeSet` `aws-marketplace:DescribeChangeSet`
Edit product	`aws-marketplace:StartChangeSet` `aws-marketplace:DescribeChangeSet`
Create custom offer	`aws-marketplace:StartChangeSet` `aws-marketplace:DescribeChangeSet`
Edit custom offer	`aws-marketplace:StartChangeSet` `aws-marketplace:DescribeChangeSet`
View product details	`aws-marketplace:DescribeEntity` `aws-marketplace:ListEntities`
View product's custom offer	`aws-marketplace:DescribeEntity`

Console action	IAM permission

View product dashboard	`aws-marketplace:ListEntities`
	
	`aws-marketplace:DescribeEntity`

List products to which a data set or revision has been published	`aws-marketplace:ListEntities`
	
	`aws-marketplace:DescribeEntity`

List subscription verification requests	`aws-marketplace:ListAgreementApprovalRequests`
Approve subscription verification requests	`aws-marketplace:AcceptAgreementApprovalRequest`
Decline subscription verification requests	`aws-marketplace:RejectAgreementApprovalRequest`
Delete information from subscription verification requests	`aws-marketplace:UpdateAgreementApprovalRequest`

Logging and Monitoring in AWS Data Exchange

Monitoring is an important part of the well-architected nature of AWS Data Exchange. You should collect monitoring data from each part of your AWS solution so that you can more easily debug a multi-point failure, if one occurs. AWS provides several tools for monitoring your resources and activity in AWS Data Exchange so you can plan for and respond to potential incidents.

The logging of actions and events in AWS Data Exchange is accomplished through its integration with Amazon CloudWatch.

The following sections describe monitoring and logging in AWS Data Exchange:

Monitoring

Monitoring is an important part of maintaining the reliability, availability, and performance of AWS Data Exchange and your other AWS solutions. AWS provides the following monitoring tools to watch AWS Data Exchange, report when something is wrong, and take automatic actions when appropriate:

- Amazon CloudWatch Events delivers a near real-time stream of system events that describe changes in AWS resources. CloudWatch Events enables automated event-driven computing, because you can write rules that watch for certain events and trigger automated actions in other AWS services when these events occur. For more information, see the Amazon CloudWatch Events User Guide.
- Amazon CloudWatch Logs makes it possible for you to monitor, store, and access your log files from Amazon EC2 instances, CloudTrail, and other sources. CloudWatch Logs can monitor information in the log files and notify you when certain thresholds are met. You can also archive your log data in highly durable storage. For more information, see the Amazon CloudWatch Logs User Guide.
- AWS CloudTrail captures API calls and related events made by or on behalf of your AWS account and delivers the log files to an Amazon S3 bucket that you specify. You can identify which users and accounts called AWS, the source IP address from which the calls were made, and when the calls occurred.

Related Topics

- Logging AWS Data Exchange API Calls with AWS CloudTrail (p. 49)

CloudWatch Events

As a subscriber with an active subscription to a product, you receive a CloudWatch event from AWS Data Exchange every time the provider publishes new revisions. The CloudWatch event contains the `DataSetId` and the list of `RevisionIds` that have been published. The detail type of the CloudWatch event is set to `Revision Published To Data Set`.

Here is an example CloudWatch event body for an updated revision:

```
{
    "version": "0",
    "id": "dc529cb6-2e23-4c5f-d020-EXAMPLE92231",
    "detail-type": "Revision Published To Data Set",
    "source": "aws.dataexchange",
    "account": "123456789012",
    "time": "2019-10-10T04:16:28Z",
    "region": "us-east-1",
    "resources": [
        "aae4c2cdEXAMPLE54f9369dEXAMPLE66"
    ],
    "detail": {
        "RevisionIds": [
            "3afc623EXAMPLE099e6fcc8EXAMPLEe7"
        ]
    }
}
```

Logging AWS Data Exchange API Calls with AWS CloudTrail

AWS Data Exchange is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in AWS Data Exchange. AWS CloudTrail captures all calls to AWS Data Exchange APIs as events, including calls from the AWS Data Exchange console and from code calls to the AWS Data Exchange API operations.

If you create a trail, you can enable continuous delivery of CloudTrail events to an S3 bucket, including events for AWS Data Exchange. If you don't configure a trail, you can still view the most recent events in the CloudTrail console in **Event history**. Using the information collected by CloudTrail, you can determine the request that was made to AWS Data Exchange, the IP address from which the request was made, who made the request, when it was made, and other details.

> **Important**
> Some actions you can take are console-only actions. There is no corresponding API in the AWS SDK or AWS CLI. These are actions that rely on AWS Marketplace functionality, such as publishing or subscribing to a product. AWS Data Exchange provides CloudTrail logs for a subset of these console-only actions. See the following list of console-only actions for which CloudTrail logs are provided.
> For more information, see What Is AWS CloudTrail?

In addition to CloudTrail events for all the AWS Data Exchange APIs and corresponding console actions, AWS Data Exchange also provides CloudTrail trails for a subset of the AWS Marketplace-backed console-only actions. AWS Data Exchange provides a CloudTrail log for the following console-only actions:

Subscriber Actions

- Subscribe to a product
- Send subscription verification request
- Enable subscription auto-renewal
- Disable subscription auto-renewal
- Cancel subscription verification request

Provider Actions

- Publish a product
- Unpublish a product
- Edit a product
- Create custom offer
- Edit custom offer
- Approve subscription verification request
- Decline subscription verification request
- Delete subscriber contact information

AWS Data Exchange Information in CloudTrail

CloudTrail is enabled when you create your AWS account. When activity occurs in AWS Data Exchange, the activity is recorded in a CloudTrail event along with other AWS service events in **Event history**. You can view, search, and download recent events in your AWS account. For more information, see Viewing Events with CloudTrail Event History in the *AWS CloudTrail User Guide*.

For an ongoing record of events in your AWS account, including events for AWS Data Exchange, create a trail. CloudTrail uses this trail to deliver log files to an S3 bucket. By default, when you use the console to create a trail, it applies to all AWS Regions. The trail logs events from all AWS Regions and delivers the log files to the S3 bucket that you specify. You can configure other AWS services to further analyze and act upon the event data collected in CloudTrail logs. For more information, see:

- Overview for Creating a Trail
- CloudTrail Supported Services and Integrations
- Configuring Amazon SNS Notifications for CloudTrail
- Receiving CloudTrail Log Files from Multiple Regions
- Receiving CloudTrail Log Files from Multiple Accounts

All AWS Data Exchange actions are logged by CloudTrail and are documented in the AWS Data Exchange API Reference. For example, calls to the `CreateDataSet`, `StartImportAssetsFromS3Workflow`, and `ListRevisionAssets` API actions generate entries in the CloudTrail log files.

Every event or log entry contains information about who generated the request. The identity information helps you determine the following:

- Whether the request was made with root or IAM user credentials.
- Whether the request was made with temporary security credentials for a role or federated user.
- Whether the request was made by another AWS service.

For more information, see CloudTrail userIdentity Element.

Understanding AWS Data Exchange Log File Entries

A trail is a configuration that makes it possible to deliver events as log files to an S3 bucket that you specify. CloudTrail log files contain one or more log entries. An event represents a single request from any source and includes information about the requested action, the date and time of the action, request parameters, and so on. CloudTrail log files are not an ordered stack trace of the public API calls, so they do not appear in any order.

Note
These examples have been formatted to improve readability. In a CloudTrail log file, all entries and events are concatenated into a single line. This example has been limited to a single AWS Data Exchange entry. In a real CloudTrail log file, you see entries and events from multiple AWS services.

The following example shows a CloudTrail log entry that demonstrates the `CreateDataSet` action:


```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE:account_name",
        "arn": "arn:aws:sts::123456789012:user/Mary_Major",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2018-06-20T18:32:25Z"
            },
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/Admin",
                "accountId": "123456789012",
                "userName": "username"
            }
        }
    },
    "eventTime": "2018-06-20T19:04:36Z",
    "eventSource": "dataexchange.amazonaws.com",
    "eventName": "CreateDataSet",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "203.0.113.12",
    "userAgent": "console.amazonaws.com",
    "requestParameters": {
        "Name": "MyDataSet",
        "AssetType": "S3_SNAPSHOT",
        "Description": "This is my dataset"
    },
    "responseElements": {
        "Origin": "OWNED",
        "AssetType": "S3_SNAPSHOT",
        "Name": "MyDataSet",
        "CreatedAt": 1726255485679,
        "UpdatedAt": 1726255485679,
        "Arn": "arn:aws:dataexchange:us-east-1:123456789012:data-sets/DataSetIdentifier",
        "Id": "DataSetIdentifier",
        "Description": "This is my dataset"
    },
    "requestID": "cb8c167e-EXAMPLE",
    "eventID": "e3c6f4ce-EXAMPLE",
    "readOnly": false,
    "eventType": "AwsApiCall",
```

```
    "recipientAccountId": "123456789012"
}>
```

Compliance Validation for AWS Data Exchange

For a list of AWS services in scope of specific compliance programs, see AWS Services in Scope by Compliance Program. For general information, see AWS Compliance Programs.

You can use AWS Artifact to download third-party audit reports. For information, see Downloading Reports in AWS Artifact in the *AWS Artifact User Guide*.

Your compliance responsibility when using AWS Data Exchange is determined by the sensitivity of your data, your company's compliance objectives, and applicable laws and regulations. AWS provides the following resources to help with compliance:

- Security and Compliance Quick Start Guides – These deployment guides discuss architectural considerations and provide steps for deploying security- and compliance-focused baseline environments on AWS.
- Architecting for HIPAA Security and Compliance Whitepaper – This whitepaper describes how companies can use AWS to create HIPAA-compliant applications.
- AWS Compliance Resources – This collection of workbooks and guides might apply to your industry and location.
- AWS Config – This AWS service assesses how well your resource configurations comply with internal practices, industry guidelines, and regulations.
- AWS Security Hub – This AWS service provides a comprehensive view of your security state within AWS that helps you check your compliance with security industry standards and best practices.

Resilience in AWS Data Exchange

The AWS global infrastructure is built around AWS Regions and Availability Zones. AWS Regions provide multiple physically separated and isolated Availability Zones, which are connected with low-latency, high-throughput, and highly redundant networking. With Availability Zones, you can design and operate applications and databases that fail over between Availability Zones without interruption. Availability Zones are more highly available, fault tolerant, and scalable than traditional single or multiple data center infrastructures.

AWS Data Exchange has a single, globally available product catalog offered by providers. Subscribers can see the same catalog, regardless of which region they are using. The resources underlying the product (data sets, revisions, assets) are regional resources that you manage programmatically or through the AWS Data Exchange console in supported AWS Regions. AWS Data Exchange replicates your data across multiple Availability Zones within the AWS Regions where the service operates. For information about supported regions, see Global Infrastructure Region Table.

For more information about AWS Regions and Availability Zones, see AWS Global Infrastructure.

Infrastructure Security in AWS Data Exchange

AWS Data Exchange is protected by the AWS global network security procedures that are described in the Amazon Web Services: Overview of Security Processes whitepaper.

You use AWS published API calls to access AWS services and resources through the network. Clients must support Transport Layer Security (TLS) 1.0 or later. We recommend TLS 1.2 or later. Clients must also support cipher suites with perfect forward secrecy (PFS), such as Ephemeral Diffie-Hellman (DHE) or Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). Most modern systems, such as Java 7 and later, support these modes.

Requests must also be signed by using an access key ID and a secret access key that is associated with an IAM principal. Or, you can use the AWS Security Token Service (AWS STS) to generate temporary security credentials to sign requests.

AWS Data Exchange and interface VPC endpoints (AWS PrivateLink)

You can establish a private connection between your VPC and AWS Data Exchange by creating an *interface VPC endpoint*. Interface endpoints are powered by AWS PrivateLink, a technology that enables you to privately access AWS Data Exchange APIs without an internet gateway, NAT device, VPN connection, or AWS Direct Connect connection. Instances in your VPC don't need public IP addresses to communicate with AWS Data Exchange APIs. Traffic between your VPC and AWS Data Exchange does not leave the Amazon network.

Each interface endpoint is represented by one or more Elastic Network Interfaces in your subnets.

For more information, see Interface VPC endpoints (AWS PrivateLink) in the *Amazon VPC User Guide*.

Considerations for AWS Data Exchange VPC endpoints

Before you set up an interface VPC endpoint for AWS Data Exchange, ensure that you review Interface endpoint properties and limitations in the *Amazon VPC User Guide*.

AWS Data Exchange supports making calls to all of its API actions from your VPC.

VPC endpoint policies are not supported for AWS Data Exchange. By default, full access to AWS Data Exchange is allowed through the endpoint. For more information, see Controlling access to services with VPC endpoints in the *Amazon VPC User Guide*.

Creating an interface VPC endpoint for AWS Data Exchange

You can create a VPC endpoint for the AWS Data Exchange service using either the Amazon VPC console or the AWS Command Line Interface (AWS CLI). For more information, see Creating an interface endpoint in the *Amazon VPC User Guide*.

Create a VPC endpoint for AWS Data Exchange using the following service name:

- com.amazonaws.*region*.dataexchange

If you enable private DNS for the endpoint, you can make API requests to AWS Data Exchange using its default DNS name for the Region, for example, `com.amazonaws.us-east-1.dataexchange`.

For more information, see Accessing a service through an interface endpoint in the *Amazon VPC User Guide*.

Creating a VPC endpoint policy for AWS Data Exchange

You can attach an endpoint policy to your VPC endpoint that controls access to AWS Data Exchange. The policy specifies the following information:

- The principal that can perform actions.
- The actions that can be performed.
- The resources on which actions can be performed.

For more information, see Controlling access to services with VPC endpoints in the *Amazon VPC User Guide*.

Example: VPC endpoint policy for AWS Data Exchange actions

The following is an example of an endpoint policy for AWS Data Exchange. When attached to an endpoint, this policy grants access to the listed AWS Data Exchange actions for all principals on all resources.

This example VPC endpoint policy allows full access only to the IAM user `bts` in AWS account `123456789012` from `vpc-12345678`. The IAM user `readUser` is allowed to read the resources, but all other IAM principals are denied access to the endpoint.

```
{
    "Id": "example-policy",
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Allow administrative actions from vpc-12345678",
            "Effect": "Allow",
            "Principal": {
                "AWS": [
                    "arn:aws:iam::123456789012:user/bts"
                ]
            },
            "Action": "*",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:sourceVpc": "vpc-12345678"
                }
            }
        },
        {
            "Sid": "Allow ReadOnly actions",
            "Effect": "Allow",
            "Principal": {
                "AWS": [
                    "arn:aws:iam::123456789012:user/readUser"
                ]
            },
            "Action": [
                "dataexchange:list*",
                "dataexchange:get*"
            ],
            "Resource": "*",
        }
    ]
}
```

Document History

The following table describes the documentation for this release of the *AWS Data Exchange Users Guide*.

update-history-change	update-history-description	update-history-date
More eligible jurisdictions (p. 55)	The following are now eligible to become sellers on AWS Marketplace: Bahrain, Norway, Switzerland, and the United Arab Emirates (UAE). For more information, see Eligible jurisdictions for AWS Data Exchange products (p. 14).	June 16, 2020
Added encryption support for exporting data sets (p. 55)	AWS Data Exchange now supports configurable encryption parameters when exporting data sets to Amazon S3. For more information, see Exporting Assets to an Amazon S3 Bucket (p. 33).	April 27, 2020
AWS Data Exchange is now generally available (p. 55)	AWS Data Exchange is a service that makes it easy for AWS customers to create, update, maintain, and securely exchange file-based data set in the AWS Cloud.	November 13, 2019

Using AWS Data Exchange with the AWS Marketplace Catalog API

You can find supplemental information for using AWS Data Exchange and the AWS Marketplace Catalog API.

Topics

The AWS Marketplace Catalog API service provides an API interface for you as a provider to programmatically access the AWS Marketplace self-service publishing capabilities.

The API aims to support a wide range of operations for you to view and manage your products. You can extend your internal build or deployment pipeline to AWS Marketplace through API integration to automate your product update process. You can also create your own internal user interface on top of the API to manage your products on the AWS Marketplace.

You can use the AWS Marketplace Catalog API to view AWS Data Exchange products and publish new revisions to them. To view your products, you can use the `ListEntities` and `DescribeEntity` APIs. To publish new revisions to your AWS Data Exchange product, you need to create a new change set, which is the Catalog API resource that represents an asynchronous operation used to manage products. For more information, see the AWS Marketplace Catalog API Reference.

In order to add revisions to your AWS Data Exchange product, you need to create a change set of type `addRevisions`. To do so, you can use the `StartChangeSet` API and specify the change type, the product identifier, the product type, and the details including the data set and revision ARNs.

Keep the following in mind when working with the Catalog API:

- Each AWS Data Exchange product is represented in the Catalog API as an https://docs.aws.amazon.com/marketplace-catalog/latest/api-reference/API_Entity.html.
- AWS Data Exchange products have `DataProduct` as the `EntityType`.
- You can update multiple products in a single `addRevisions` change set.
- Each change set is scoped to a single data set within a product. If your product has more than one data set and you need to update all of them, create a separate change set for each data set.
- Each product can have only one concurrently running change set at a time. This means that you can't create a second change set until the first one has stopped running.

AddRevisions Exceptions

The following exceptions can occur when you use the`AddRevisions` change type with AWS Data Exchange:

REVISION_NOT_FOUND

This happens when the requested resource was not found. To resolve this issue, make sure that there's not a typo in the revision ARN and that your AWS account owns the resource, and try again.

REVISION_NOT_FINALIZED

Revisions must be finalized prior to being added to AWS Data Exchange products. To resolve this issue, ensure that the revisions with your specified ARNs are finalized, and try again.

DATA_SET_NOT_FOUND

This happens when the requested data set was not found. To resolve this issue, ensure that there's not a typo in the data set ARN and that your AWS account owns the data set, and try again.

INVALID_INPUT

The request couldn't be processed due to invalid input. To resolve this issue, ensure that there's not a typo in the request and that the list of revisions has at least one and no more than five revisions.

DATA_SET_NOT_PUBLISHED

The requested resource has not been published in this product. To resolve this issue, ensure that there's not a typo in the ARN(s) for the data set(s). You can also publish a new product that includes those data set(s).

REVISION_DUPLICATE_PROVIDED

This happens when the same revision request occurs more than once. To resolve this issue, ensure that the revisions aren't duplicates, and try again.

Tutorial: Adding New Data Set Revisions to a Published Data Product

This tutorial walks you through detailed steps to publish new AWS Data Exchange data set revisions to an existing product. The tutorial has the following high-level steps.

Topics

Set Up IAM Permissions

Before you begin, you need IAM permissions for using the AWS Marketplace Catalog API. These permissions are in addition to the permissions you need for using AWS Data Exchange.

1. Navigate your browser to the IAM console and sign in using an AWS account that can manage IAM permissions.
2. From the left navigation pane, choose **Policies**.
3. Choose **Create policy**.
4. Choose the **JSON** tab, and provide the following permissions. This provides full access to the AWS Marketplace Catalog API. You can restrict access as appropriate for your use case.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:CancelChangeSet",
        "aws-marketplace:ListChangeSets",
        "aws-marketplace:DescribeEntity",
        "aws-marketplace:StartChangeSet",
        "aws-marketplace:ListEntities",
        "aws-marketplace:DescribeChangeSet"
      ],
      "Resource": "*"
    }
  ]
}
```

5. Choose **Review policy**.
6. Provide a name for the policy (for example, **`CatalogAPIFullAccess`**), and then choose **Create Policy**.
7. Using the IAM console, choose the users, groups, or roles that you want to attach the policy to.

Access the AWS Marketplace Catalog API

To access the AWS Marketplace Catalog API, use the following HTTP client endpoint.

```
catalog.marketplace.us-east-1.amazonaws.com
```

Get Your Product ID from the AWS Data Exchange Console

Before you can use the AWS Marketplace Catalog API to publish new revisions, get your product ID from the AWS Data Exchange console. Navigate to the **Product Dashboard**, and then copy the product you would like to publish revisions for.

Describe Product Details

Before you add revisions to a data set in your data product, you must make a `DescribeEntity` call with the ID of your product. The output contains the product details and an `EntityIdentifier` attribute, which is the entity ID (your product ID) appended with a `revision number` (for example, `@9`). The revision number reflects the most recent edit or change made to your published product.

Sample Request

```
https://catalog.marketplace.us-east-1.amazonaws.com/DescribeEntity?
catalog=AWSMarketplace&entityId=entity-id
```

Sample Response

```
{
    "Details": "{\"Title\":\"Example Data Product\",\"Presentation\":{\"ShortDescription
\":\"Descriptive text that appears on the tiles in the Product catalog page\",
```

```
\"FullDescription\":\"Descriptive text that appears on the product's detail page after the
 product is published. \",\"Logo\":\"logo-url\",\"Highlights\":[]},\"Lifecycle\":\"Public
\",\"ProductCode\":\"product-code\",\"DataSets\":[{\"ProductDataSetId\":\"product-data-
set-id\",\"DataSetArn\":\"data-set-arn\",\"Name\":\"Example Data Set\",\"Description\":
\"Example Data Set\",\"CreationDate\":\"2019-09-08T16:10:33.011Z\",\"LastRevisionAddedDate
\":\"2019-09-08T16:10:33.011Z\",\"PublishedDataSetArn\":\"published-data-set-arn\"}]}",
    "EntityArn": "arn:aws:aws-marketplace:us-east-1:account-id:AWSMarketplace/
DataProduct/entity-id",
    "EntityIdentifier": "entity-id@1",
    "EntityType": "DataProduct@1.0",
    "LastModifiedDate": "2019-09-08T16:10:38.287Z"
}
```

Note
If you are using an HTTP client, you must unescape the JSON output to view the data in a nested structure. You might also need to reform the output to suit your needs.

Start a Change Request

To start a change request to add revisions to a data set in your test product:

1. Copy the `DescribeEntity` response from Describe Product Details (p. 58) and include the following elements:
   - Entity type with the version number (for example, `DataProduct@1.0`).
   - Identifier with the revision number (for example, `product-test@1`).
   - The `DataSetArn` for the data set to which you would like to add revisions (for example, `arn:aws:dataexchange:us-east-1:account-id:data-sets/data-set-id`).
2. Make a `StartChangeSet` request with an `AddRevisions` change type. The details of the `AddRevisions` change object, in the request body, should contain the following:
   - `DataSetArn` should be the data set to which you'd like to add revisions.
   - `RevisionArns` should be a list of the revisions that you want to publish to the data set in the product. For more information about the number of revisions that a single change can include, see AWS Data Exchange Limits (p. 35).

Sample Request

```
https://catalog.marketplace.us-east-1.amazonaws.com/StartChangeSet
```

Sample Request Body

```
{
  "Catalog": "AWSMarketplace",
  "ChangeSetName": "Adding revisions to my test Data Product",
  "ChangeSet": [
    {
      "ChangeType": "AddRevisions",
      "Entity": {
        "Identifier": "entity-id@1",
        "Type": "DataProduct@1.0"
      },
      "Details": "{\"DataSetArn\": \"data-set-arn\", \"RevisionArns\": [\"revision-arn\",
 \"revision-arn-2\"] }"
    }
  ]
}
```

Sample Response

```
{
  "ChangeSetId": "cs-bnEXAMPLE4mkz9oh",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:account-id:AWSMarketplace/
ChangeSet/cs-bnEXAMPLE4mkz9oh"
}
```

Check the Status of Your Change Set

After you use the `StartChangeSet` API to start the change request, you can use the `DescribeChangeSet` API to check its status. Provide the change set ID returned in the `StartChangeSet` API response.

Sample Request

```
https://catalog.marketplace.us-east-1.amazonaws.com/DescribeChangeSet?
catalog=AWSMarketplace&changeSetId=cs-bnEXAMPLE4mkz9oh
```

Sample Request Body

```
{
"changeSetId":"cs-bnEXAMPLE4mkz9oh"
}
```

Sample Response

```
{
    "ChangeSetId": "cs-bnEXAMPLE4mkz9oh",
    "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:account-id:AWSMarketplace/
ChangeSet/cs-bnEXAMPLE4mkz9oh",
    "ChangeSetName": "Adding revisions to my test Data Product",
    "StartTime": "2018-09-20T19:45:03.115+0000",
    "EndTime": "2018-09-20T19:48:12.517+0000",
    "Status": "SUCCEEDED",
    "FailureDescription": null,
    "ChangeSet": [
        {
            "ChangeType": "AddRevisions",
            "Entity": {
                "Type": "DataProduct@1.0",
                "Identifier": "entity-id@1"
            },
            "ErrorList": []
        }
    ]
}
```

AWS glossary

For the latest AWS terminology, see the AWS glossary in the *AWS General Reference*.