AWS Marketplace Catalog API

Reference
# Table of Contents

AWS Marketplace Catalog API .................................................................................................................. 1
Supported AWS Regions .......................................................................................................................... 1
Catalog API entities ................................................................................................................................. 1
  General structure ................................................................................................................................. 1
  Type versioning ................................................................................................................................. 1
  Identifier ........................................................................................................................................... 2
  Facets ............................................................................................................................................... 2
  Entity type ....................................................................................................................................... 3
Working with change sets ..................................................................................................................... 3
  Making multiple change requests simultaneously .............................................................................. 4
Working with the Details attribute ...................................................................................................... 5
Using DescribeEntity ............................................................................................................................. 6
API access control ............................................................................................................................... 10
  Allowing actions with AWS managed policies ................................................................................... 10
  Allowing actions on all resources ..................................................................................................... 11
  Allowing actions on specific resources ............................................................................................ 11
  Allowing actions with specific ChangeType condition key ............................................................... 12
  Allowing actions with specific aws:ResourceTag condition key ...................................................... 13
  Managing tags on resources ............................................................................................................. 15
  Managing tags when requesting changes to resources ..................................................................... 17
  Granting permission to manage tags on resources ............................................................................ 18
  Granting permission to manage tags on resources only when those resources have specific tags ....................................................................................................................................................... 18
  Granting permission to create entities and change sets only with tags ............................................ 19
Service quotas ..................................................................................................................................... 19
Working with seller products ............................................................................................................... 21
  Finding your product ID (server products) ....................................................................................... 21
Change set status and errors .................................................................................................................. 21
Working with single-AMI products ...................................................................................................... 26
  Updating product details for an AMI-based product ....................................................................... 27
  Adding a new version to an AMI-based product .............................................................................. 28
  Updating version information for an AMI-based product ............................................................... 31
  Restricting a version of an AMI-based product .............................................................................. 33
  Errors in the AMI products API ....................................................................................................... 34
Working with container-based products .............................................................................................. 38
  Updating product information for a container-based product ......................................................... 38
  Adding a new version to a container-based product ....................................................................... 40
  Creating repositories and resources for a container-based product .............................................. 43
  Updating version information for a container-based product ........................................................... 45
  Restricting a version of a container-based product ......................................................................... 48
  Errors in the container products API ............................................................................................... 49
Working with a private marketplace ...................................................................................................... 56
  Creating a private marketplace ........................................................................................................ 56
  Changing the branding of a private marketplace experience ............................................................ 59
  Enabling or disabling a private marketplace experience ................................................................. 61
  Enabling or disabling user requests ................................................................................................ 61
  Getting a list of products in a private marketplace experience ....................................................... 62
  Adding or removing products from a private marketplace .............................................................. 64
Finding products .................................................................................................................................. 65
Working with private marketplaces for AWS Organizations ................................................................. 65
  Associating accounts to experiences ............................................................................................... 66
  Archiving and reactivating a private marketplace experience ....................................................... 68
  Errors in the private marketplace API ............................................................................................. 69
Entity types defined by private marketplace ...................................................................................... 70
Working with AWS RAM to share resources ....................................................................................... 71
Prerequisites for sharing AWS Marketplace entities ................................................................. 71
Sharing an AWS Marketplace entity .......................................................................................... 71
Differences between sharing an entity through AWS RAM and the AWS Marketplace Catalog
API ........................................................................................................................................ 72
Attach read-only policy to your resource ............................................................................... 73
Attach read and write resource policy to your resource ...................................................... 73
View resource policy set on your resource ............................................................................ 74
Delete resource policy on your resource ............................................................................... 75
View all resources owned by you and shared with you ............................................................ 75
Logging and notifications .......................................................................................................... 77
Logging with AWS CloudTrail ................................................................................................. 77
AWS Marketplace Catalog API information in CloudTrail ..................................................... 77
Understanding AWS Marketplace catalog log file entries ....................................................... 78
Amazon EventBridge notifications ......................................................................................... 79
AWS Marketplace Discovery API ............................................................................................. 80
Getting access to the Discovery API ....................................................................................... 80
Logging Discovery API calls with CloudTrail ....................................................................... 80
Discovery API information in CloudTrail ............................................................................. 81
Understanding Discovery API log file entries ....................................................................... 81
Release notes for AWS Marketplace Discovery API ............................................................. 82
Discovery API release notes for 2022 ................................................................................... 82
AWS Marketplace Agreements API ......................................................................................... 84
Logging Agreements API calls with CloudTrail .................................................................... 84
Agreements API information in CloudTrail .......................................................................... 81
Understanding Agreements API actions ................................................................................. 85
Understanding Agreements API log file entries ..................................................................... 85
Actions ..................................................................................................................................... 87
CancelChangeSet ...................................................................................................................... 88
Request Syntax ....................................................................................................................... 88
URI Request Parameters ........................................................................................................ 88
Request Body .......................................................................................................................... 88
Response Syntax .................................................................................................................... 88
Response Elements ............................................................................................................... 88
Errors ...................................................................................................................................... 89
See Also ................................................................................................................................. 90
DeleteResourcePolicy ............................................................................................................ 91
Request Syntax ....................................................................................................................... 91
URI Request Parameters ........................................................................................................ 91
Request Body .......................................................................................................................... 91
Response Syntax .................................................................................................................... 91
Response Elements ............................................................................................................... 91
Errors ...................................................................................................................................... 91
See Also ................................................................................................................................. 92
DescribeChangeSet .................................................................................................................. 93
Request Syntax ....................................................................................................................... 93
URI Request Parameters ........................................................................................................ 93
Request Body .......................................................................................................................... 93
Response Syntax .................................................................................................................... 93
Response Elements ............................................................................................................... 93
Errors ...................................................................................................................................... 94
See Also ................................................................................................................................. 95
DescribeEntity ......................................................................................................................... 97
Request Syntax ....................................................................................................................... 97
URI Request Parameters ........................................................................................................ 97
Request Body .......................................................................................................................... 97
Response Syntax .................................................................................................................... 97
Response Elements ............................................................................................................... 97
<table>
<thead>
<tr>
<th>Function</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Errors</td>
<td>98</td>
</tr>
<tr>
<td>See Also</td>
<td>99</td>
</tr>
<tr>
<td>GetResourcePolicy</td>
<td>100</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>100</td>
</tr>
<tr>
<td>URI Request Parameters</td>
<td>100</td>
</tr>
<tr>
<td>Request Body</td>
<td>100</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>100</td>
</tr>
<tr>
<td>Response Elements</td>
<td>100</td>
</tr>
<tr>
<td>Errors</td>
<td>101</td>
</tr>
<tr>
<td>See Also</td>
<td>101</td>
</tr>
<tr>
<td>ListChangeSets</td>
<td>102</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>102</td>
</tr>
<tr>
<td>URI Request Parameters</td>
<td>102</td>
</tr>
<tr>
<td>Request Body</td>
<td>102</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>103</td>
</tr>
<tr>
<td>Response Elements</td>
<td>103</td>
</tr>
<tr>
<td>Errors</td>
<td>104</td>
</tr>
<tr>
<td>See Also</td>
<td>104</td>
</tr>
<tr>
<td>ListEntities</td>
<td>106</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>106</td>
</tr>
<tr>
<td>URI Request Parameters</td>
<td>106</td>
</tr>
<tr>
<td>Request Body</td>
<td>106</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>107</td>
</tr>
<tr>
<td>Response Elements</td>
<td>108</td>
</tr>
<tr>
<td>Errors</td>
<td>108</td>
</tr>
<tr>
<td>See Also</td>
<td>109</td>
</tr>
<tr>
<td>ListTagsForResource</td>
<td>110</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>110</td>
</tr>
<tr>
<td>URI Request Parameters</td>
<td>110</td>
</tr>
<tr>
<td>Request Body</td>
<td>110</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>110</td>
</tr>
<tr>
<td>Response Elements</td>
<td>110</td>
</tr>
<tr>
<td>Errors</td>
<td>111</td>
</tr>
<tr>
<td>See Also</td>
<td>111</td>
</tr>
<tr>
<td>PutResourcePolicy</td>
<td>113</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>113</td>
</tr>
<tr>
<td>URI Request Parameters</td>
<td>113</td>
</tr>
<tr>
<td>Request Body</td>
<td>113</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>113</td>
</tr>
<tr>
<td>Response Elements</td>
<td>113</td>
</tr>
<tr>
<td>Errors</td>
<td>114</td>
</tr>
<tr>
<td>See Also</td>
<td>114</td>
</tr>
<tr>
<td>StartChangeSet</td>
<td>115</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>115</td>
</tr>
<tr>
<td>URI Request Parameters</td>
<td>115</td>
</tr>
<tr>
<td>Request Body</td>
<td>115</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>116</td>
</tr>
<tr>
<td>Response Elements</td>
<td>117</td>
</tr>
<tr>
<td>Errors</td>
<td>117</td>
</tr>
<tr>
<td>Examples</td>
<td>117</td>
</tr>
<tr>
<td>See Also</td>
<td>118</td>
</tr>
<tr>
<td>TagResource</td>
<td>120</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>120</td>
</tr>
<tr>
<td>URI Request Parameters</td>
<td>120</td>
</tr>
<tr>
<td>Request Body</td>
<td>120</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>120</td>
</tr>
<tr>
<td>Response Elements</td>
<td>120</td>
</tr>
</tbody>
</table>
## AWS Marketplace Catalog API Reference

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Errors</td>
<td>121</td>
</tr>
<tr>
<td>See Also</td>
<td>121</td>
</tr>
<tr>
<td>UntagResource</td>
<td>123</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>123</td>
</tr>
<tr>
<td>URI Request Parameters</td>
<td>123</td>
</tr>
<tr>
<td>Request Body</td>
<td>123</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>123</td>
</tr>
<tr>
<td>Response Elements</td>
<td>124</td>
</tr>
<tr>
<td>Errors</td>
<td>124</td>
</tr>
<tr>
<td>See Also</td>
<td>124</td>
</tr>
<tr>
<td>Data Types</td>
<td>126</td>
</tr>
<tr>
<td>Change</td>
<td>127</td>
</tr>
<tr>
<td>Contents</td>
<td>127</td>
</tr>
<tr>
<td>See Also</td>
<td>128</td>
</tr>
<tr>
<td>ChangeSetSummaryListitem</td>
<td>129</td>
</tr>
<tr>
<td>Contents</td>
<td>129</td>
</tr>
<tr>
<td>See Also</td>
<td>130</td>
</tr>
<tr>
<td>ChangeSummary</td>
<td>131</td>
</tr>
<tr>
<td>Contents</td>
<td>131</td>
</tr>
<tr>
<td>See Also</td>
<td>132</td>
</tr>
<tr>
<td>Entity</td>
<td>133</td>
</tr>
<tr>
<td>Contents</td>
<td>133</td>
</tr>
<tr>
<td>See Also</td>
<td>133</td>
</tr>
<tr>
<td>EntitySummary</td>
<td>134</td>
</tr>
<tr>
<td>Contents</td>
<td>134</td>
</tr>
<tr>
<td>See Also</td>
<td>135</td>
</tr>
<tr>
<td>ErrorDetail</td>
<td>136</td>
</tr>
<tr>
<td>Contents</td>
<td>136</td>
</tr>
<tr>
<td>See Also</td>
<td>136</td>
</tr>
<tr>
<td>Filter</td>
<td>137</td>
</tr>
<tr>
<td>Contents</td>
<td>137</td>
</tr>
<tr>
<td>See Also</td>
<td>137</td>
</tr>
<tr>
<td>Sort</td>
<td>139</td>
</tr>
<tr>
<td>Contents</td>
<td>139</td>
</tr>
<tr>
<td>See Also</td>
<td>139</td>
</tr>
<tr>
<td>Tag</td>
<td>140</td>
</tr>
<tr>
<td>Contents</td>
<td>140</td>
</tr>
<tr>
<td>See Also</td>
<td>140</td>
</tr>
<tr>
<td>Common Parameters</td>
<td>141</td>
</tr>
<tr>
<td>Common Errors</td>
<td>143</td>
</tr>
<tr>
<td>Document history</td>
<td>145</td>
</tr>
</tbody>
</table>
AWS Marketplace Catalog API

The AWS Marketplace Catalog API service provides an API interface to manage AWS Marketplace for your AWS organization or AWS account. For approved sellers, you can manage your products programmatically, including the self-service publishing capabilities on the AWS Marketplace Management Portal. For private marketplace administrators, you can manage your private marketplace programmatically.

With Catalog API actions, you can view and update your existing product programmatically. You can automate your product update process by integrating the AWS Marketplace Catalog API with your AWS Marketplace product build or deployment pipelines. You can also create your own applications on top of the Catalog API to manage your products in AWS Marketplace. You can manage the products that users in your AWS account or AWS organization can see and purchase through your private marketplace.

The AWS Marketplace Catalog API service provides standard AWS API functionality. You can directly use the REST API actions described in Actions, or you can use an AWS SDK to access an API that's tailored to the programming language or platform that you're using. For more information about AWS application development, see Getting Started with AWS. For more information about using AWS SDKs, see AWS SDKs.

Supported AWS Regions

You can access the AWS Marketplace Catalog API from the US East (N. Virginia) AWS Region with the following endpoint.

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

Catalog API entities

AWS Marketplace entities are containers of data which serve different business purposes, such as a product or offer. Entities are categorized by types. Each entity type encapsulates data related to a specific business domain (for example, a product or a seller account).

To simplify this paradigm, entities are designed with some level of commonality in their structures. As a result, introducing a new business domain doesn’t require that you learn a completely new structure.

General structure

The general structure of any entity is:

- A named type with a version
- An identifier for the specific instance of the type
- One or more facets that include the attributes of the entity

Type versioning

Every named type has a type and version associated with it, for example, EntityProduct@1.0. The type (EntityProduct) represents the classification of the content. The version (1.0) represents the structure of EntityProduct.
The version gives you details about the structure of the entity. The following describes when a version will be changed:

- Existing entities won't be restructured without changing the version. Additions of optional new fields will result in a minor version update.
- Any feature that fundamentally changes the structure of a type leads to a major version update. Examples include:
  - Removing a field
  - Renaming a field (different name for the same semantic)
  - Changing the semantic of an existing field (for example, changing the expected type)
- A major version update can retain a subset of facets from the previous version.
- Users are provided notifications and documentation for new versions.

**Identifier**

Each entity represents a unique thing within a business domain. To identify the unique thing, we use an identifier associating an EntityId with a RevisionId, for example, `prod-ad8EXAMPLE651@3`. In this example, the EntityId is `prod-ad8EXAMPLE651` and the RevisionId is 3. Every successful change request to the entity will update the revision.

The following are important details about the identifier:

- Each entity is uniquely identified by its EntityId, which is the key to globally distinguish one entity from another.
- Each published revision of an entity has a RevisionId. The RevisionId, along with the EntityId, distinguish one published revision from another.
- AWS Marketplace generates EntityIds and RevisionIds.

You can use the DescribeEntity action to find the details and the Identifier with the most recent revisionId.

The RevisionId is an optional part of requests to StartChangeSet (see [Working with change sets](#)). If you include a RevisionId, then the request to StartChangeSet will fail with a ValidationException if the RevisionId is not the latest revision of the entity. This allows you to implement optimistic locking in your application.

**Note**

When you include a RevisionId that is not the latest revision, the ValidationException message includes the latest RevisionId.

If you omit the RevisionId, the request is performed on the latest revision of the entity automatically.

**Warning**

Two requests to change the same object could result with one request overwriting the changes of the other request, as the second request rewrites data changed by the first request. Using RevisionIds in your requests prevents this issue by not allowing a change to an earlier revision to overwrite the current revision.

**Facets**

A facet is a logical grouping of attributes. An entity usually includes several facets which represent different aspects of the entity. The attributes within a facet have the following properties:

- Each attribute has a unique name within the scope of the container it belongs to.
• Attributes can be of a simple type (string, integer, or floating number).
• Attributes can be of a complex type (container/structure or array).

**Entity type**

The entity type defines what the entity represents. An entity can be a seller product in AWS Marketplace or a private marketplace experience. For more information, see [Working with seller products](#) and [Working with a private marketplace](#).

**Working with change sets**

When using the Catalog API, requests are created and updated through entities and completed by using change requests. Every change specifies the entity to be changed, the type of change to be performed, and details of the change. The type of change to be performed is called a ChangeType. A collection of ChangeTypes is called a ChangeSet.

There are four actions that allow you to work with change sets:

- **StartChangeSet** – Requests a set of changes. The changes are added to a queue and processed. For more information, see [Working with seller products](#) and [Working with a private marketplace](#).
- **DescribeChangeSet** – Gets the details of a set of changes, including the status of the request. The statuses include:
  - PREPARING – Getting ready to apply the changes.
  - APPLYING – In the process of making the requested changes.
  - SUCCEEDED – Request was completed successfully.
  - CANCELED – Request was canceled by the user.
  - FAILED – Request was completed unsuccessfully. Further details are available in the response.
- **ListChangeSets** – Gets a list of the change sets that are currently in process.
- **CancelChangeSet** – Requests a change set be canceled. Changes can only be canceled while in the PREPARING status.

A typical workflow is to request a change with **StartChangeSet**, and then use the returned ChangeSetId to poll the **DescribeChangeSet** action until the change is complete.

The following is an example of the **DescribeChangeSet** response.

```json
{
  "ChangeSet":
  {
    "ChangeName": "myChangeName",
    "ChangeType": "UpdateInformation",
    "Details": {"ProductTitle": "My Product Title", "ShortDescription": "My product short description.", "LongDescription": "My product longer description.", "Sku": "123example456", "SupportDescription": "Need help? Contact our experts at support@example.com\n\nYour purchase includes 24x7 support.", "Categories": ["Operating Systems", "Network Infrastructure", "Application Development"]}
},
"DetailsDocument":
{
  "ProductTitle": "My Product Title",
  "ShortDescription": "My product short description."
}
```
"LongDescription": "My product longer description.",
"Sku": "123example456",
"SupportDescription": "Need help? Contact our experts at support@example.com"

Your purchase includes 24x7 support."

"Categories": [
   "Operating Systems",
   "Network Infrastructure",
   "Application Development"
]

"Entity": {
   "Identifier": "example1-abcd-1234-5ef6-7890abcdef12",
   "Type": "AmiProduct@1.0"
},
"ErrorDetailList": []

"ChangeSetArn": "arn:aws:aws-marketplace:[exampleARN]",
"ChangeSetId": "example123456789012abcdef",
"ChangeSetName": "myChangeSetName",
"EndTime": "2023-03-03T00:00:00Z",
"FailureCode": null,
"FailureDescription": null,
"StartTime": "2023-03-02T00:00:00Z",
"Status": "SUCCEEDED"

**Note**
When polling or working with change sets programmatically, you must adhere to the [Service quotas](p. 19).

After your change is complete, you can use ListEntities to find the entity that you created or modified (and its associated EntityID). You can then use DescribeEntity with the EntityID to get details about it.

For more information about working with change requests in the console for sellers, see [Creating a change request](#) in the *AWS Marketplace Seller Guide.*

### Making multiple change requests simultaneously

Within a **single change set**, you can bundle all change types and they are run together. Catalog API is built to make multiple changes simultaneously to provide the best performance. Sellers and Channel Partners can invoke changes with multiple ChangeTypes bundled into a ChangeSet. You can invoke multiple changes on single or different entities in the same ChangeSet. Catalog API evaluates which order the changes need to be applied and makes those changes.

However, if the requests are made as **separate change sets**, AWS Marketplace can't initiate conflicting change requests on the same product. In these cases, AWS Marketplace returns a ResourceInUseException error.

- For modifying AMI and container products, most changes can be made without error, with the following exceptions:
  - If two requests are the same ChangeType on the same product, the second request returns an error.
  - If one request is to update the version information, and the other request is to restrict or add a version, then the second request returns an error.
  - If a request is PREPARING, another request can be made on the same product. However, a change that is currently APPLYING may block other requests, returning an error.


For other product types and private marketplaces, you can only have a single request for a product at a time. If a different request to update the same product is made while a first request is ongoing, the second returns an error.

If there is a request for any product that is pending with the AWS Marketplace Seller Operations team, then any other requests on that product return an error.

If you receive a ResourceInUseException error for a change request, you can retry the request later. Depending on the state of the ongoing request, you can also cancel the first request, to allow the resubmitted second request to complete sooner.

Working with the Details attribute

**Note**

This section describes the legacy Details attribute in your change request, which requires additional formatting for your change details. We recommend using the alternative DetailsDocument attribute. It doesn't require additional formatting and the change details don't need to be changed. For examples of the DetailsDocument attribute, see [Working with seller products](#) and [Working with a private marketplace](#).

The Details attribute of the StartChangeSet operation is a string value. Its contents are JSON objects. To put a JSON object into a string attribute, you must convert the object to a single-line string by escaping all JSON control characters, and removing line breaks.

For example, if you are using the StartChangeSet operation with UpdateProcurementPolicy to disable requests from users in your private marketplace, make a request like the following.

```plaintext
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateProcurementPolicy",
      "Details": "<string>",
      "Entity": {
        "Type": "Experience@1.0",
        "Identifier": "exp-1234example@5"
      }
    }
  ]
}
```

In this case, the JSON object that you use for the Details attribute looks like the following (before conversion to a string).

```json
{
  "Configuration": {
    "PolicyResourceRequests": "Deny"
  }
}
```

But the Details attribute requires a string, not JSON. After converting this JSON object to a single line string, it looks like the following.

```
"{"Configuration": {"PolicyResourceRequests": "Deny"}}"
```
With this string, you can create the full change set request, as follows.

```json
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
    "Catalog": "AWSMarketplace",
    "ChangeSet": [
        {
            "ChangeType": "UpdateProcurementPolicy",
            "Details": "{"Configuration" : {"PolicyResourceRequests" : "Deny"}}",
            "Entity": {
                "Type": "Experience@1.0",
                "Identifier": "exp-1234example@5"
            }
        }
    ]
}
```

Generally, examples in this API reference show the JSON object already converted to a string. In some cases, more complicated samples with new lines are included to enhance understanding.

### Automate converting JSON to a string

Converting a JSON object to a string can be automated using tools such as `jq`, a lightweight command-line JSON processor. The following example shows using `jq` to convert a JSON object to a string that can be used in the `Details` attribute.

```bash
DETAILS_JSON='{
    "ProductTitle": "My Product Title",
    "ShortDescription": "My product short description.",
    "LongDescription": "My product long description."
}';
DETAILS_JSON_STRING="$(echo "$DETAILS_JSON" | jq 'tostring');"
```

If you echo `"${DETAILS_JSON_STRING}"`, the result is the following string with JSON properly escaped: `{"ProductTitle": "My Product",
"ShortDescription": "My product short description.",
"LongDescription": "My product long description."}

### Using DescribeEntity to get information about your entities

You can programmatically get information about your existing entities, including products and private marketplace, through the Catalog API.

The `ListEntities` action returns a list of entities. Then, you can use the `DescribeEntity` action to get details about an individual entity. This can be directly useful, for example, to catalog the products you sell. It can also be useful when updating entities, because you can get the current state of the entity before updating just the parts that you want to update.

The following example shows using `ListEntities` to get a list of container products, and then using `DescribeEntity` to get information about one of the specific products.

```json
POST /ListEntities HTTP/1.1
Content-type: application/json

```
For the entity type, you must use the entity type without the version. It returns all entities of that type (and doesn't filter on version).

Here is a sample of the response to the ListEntities action.

```json
{
  "EntitySummaryList": [
    {
      "Name": "Container Product 1",
      "EntityType": "ContainerProduct",
      "EntityId": "example1-abcd-1234-5ef6-7890abcdef12",
      "EntityArn": "arn:aws:aws-marketplace:[exampleARN]",
      "LastModifiedDate": "2021-03-01T00:00:00Z",
      "Visibility": "Public"
    },
    {
      "Name": "Container Product 2",
      "EntityType": "ContainerProduct",
      "EntityId": "example2-abcd-1234-5ef6-7890abcdef12",
      "EntityArn": "arn:aws:aws-marketplace:[exampleARN]",
      "LastModifiedDate": "2021-03-02T00:00:00Z",
      "Visibility": "Public"
    }
  ],
  "NextToken": "exampleabcdef12345..."
}
```

To get the details of one of these products, use the DescribeEntity action. The following example shows how to get details about the first product returned above.

```
GET /DescribeEntity?catalog=AWSMarketplace&entityId=example1-abcd-1234-5ef6-7890abcdef12
HTTP/1.1
```

The following shows the response to DescribeEntity.

```json
{
  "EntityType": "ContainerProduct@1.0",
  "EntityIdentifier": "example1-abcd-1234-5ef6-7890abcdef12@9",
  "EntityArn": "arn:aws:aws-marketplace:[exampleARN]",
  "LastModifiedDate": "2021-03-02T20:19:14Z",
  "Details": {"\"Versions\": [{"\"Id\": "example2-0000-aaaa-5ef6-7890abcdef12", "\"ReleaseNotes\": "My release notes", "\"UpgradeInstructions\": "N/A", "\"VersionTitle\": "1.0", "\"CreationDate\": "2021-03-02T00:00:00Z", "\"Sources\": [{"\"Type\": "DockerImages", "\"Id\": "example3-1111-bbbb-5ef6-7890abcdef12", "\"Images\": ["709825985650.dkr.ecr.us-east-1.amazonaws.com/some-seller-prefix/my-repo-1:my-tag"]}, "\"Compatibility\": {"\"Platforms\": ["Linux"]}], "\"DeliveryOptions\": [{"\"Id\": "example4-2222-cccc-2222-cccccccccccc", "\"Type\": "ElasticContainerRegistry", "\"SourceId\": "example3-1111-bbbb-5ef6-7890abcdef12", "\"Title\": "New delivery option 1", "\"ShortDescription": "Delivery option 1", "\"isRecommended": false, "\"Compatibility": {"\"AWSServices": ["ECS\", "EKS"], "\"Instructions": {"\"Usage": "test"}, "\"Recommendations": {"\"AdditionalArtifacts": [], "\"Visibility": "Limited"}}}], "\"Description\": {"\"Highlights": ["Some highlight"], "\"LongDescription": "Description of my product", "\"ProductCode": "123456789012abcdef1234567", "\"Manufacturer": null, "\"Visibility": "Limited"}, "\"AssociatedProducts": null, "\"Sku": null, "\"SearchKeywords": ["some keyword"]}, "\"ProductName": "Container Product 1", "\"ShortDescription": "Description of my product", "\"Categories": ["Operating Systems"]}, "\"PromotionalResources": {"\"LogoUrl": "https://awsmp-logos.s3.amazonaws.com/PLACEHOLDER_Logo_for_Containers_products.png"}
```
"AdditionalResources": [], "Videos": [], "SupportInformation": {
  "Description": "Description of support information., "Resources": [], "RegionAvailability": {
},
"DetailsDocument": {
  "Versions": [
    {
      "Id": "example2-0000-aaaa-5ef6-7890abcdef12",
      "ReleaseNotes": "My release notes",
      "UpgradeInstructions": "N/A",
      "VersionTitle": "1.0",
      "CreationDate": "2021-03-02T00:00:00.000Z",
      "Sources": [
        {
          "Type": "DockerImages",
          "Id": "example3-1111-bbbb-5ef6-7890abcdef12",
          "Images": [
            "709825985650.dkr.ecr.us-east-1.amazonaws.com/some-seller-prefix/my-repo-1:some-tag"
          ],
          "Compatibility": {
            "Platform": "Linux"
          }
        }
      ],
      "DeliveryOptions": [
        {
          "Id": "example4-2222-cccc-2222-ccccc-ccccc",
          "Type": "ElasticContainerRegistry",
          "SourceId": "example3-1111-bbbb-5ef6-7890abcdef12",
          "Title": "New delivery option 1",
          "ShortDescription": "Delivery option 1",
          "isRecommended": false,
          "Compatibility": {
            "AWSServices": [
              "ECS",
              "EKS"
            ],
            "Instructions": {
              "Usage": "test"
            },
            "Recommendations": {
              "AdditionalArtifacts": [],
              "Visibility": "Limited"
            }
          }
        }
      ],
      "Description": ""
Using DescribeEntity

```json
{
  " Highlights": [
    "Some highlight"
  ],
  "LongDescription": "Description of my product",
  "ProductCode": "123456789012abcdef1234567",
  "Manufacturer": null,
  "Visibility": "Limited",
  "AssociatedProducts": null,
  "Sku": null,
  "SearchKeywords": [
    "some keyword"
  ],
  "ProductTitle": "Container Product 1",
  "ShortDescription": "Description of my product",
  "Categories": [
    "Operating Systems"
  ],
  "PromotionalResources": {
    "LogoUrl": "https://awsmp-logos.s3.amazonaws.com/PLACEHOLDER_Logo_for_Containers_products.png",
    "AdditionalResources": [],
    "Videos": []
  },
  "SupportInformation": {
    "Description": "Description of support information.",
    "Resources": []
  },
  "RegionAvailability": {
    "Regions": [
      "ap-south-1",
      "eu-west-3",
      "eu-north-1",
      "eu-west-2",
      "eu-west-1",
      "ap-northeast-2",
      "ap-northeast-1",
      "me-south-1",
      "ca-central-1",
      "sa-east-1",
      "ap-east-1",
      "ap-southeast-1",
      "ap-southeast-2",
      "eu-central-1",
      "us-east-1",
      "us-east-2",
      "us-west-1",
      "us-west-2"
    ],
    "FutureRegionSupport": null
  },
  "Repositories": [
    {
      "Url": "709825985650.dkr.ecr.us-east-1.amazonaws.com/some-seller-prefix/my-repo-1",
    }
  ]
}
```
API access control

You can use the AWS Marketplace Catalog API to manage a seller product in AWS Marketplace or an experience in a private marketplace. However, first make sure your user or role can access the API functionality that you want to call.

Use AWS Identity and Access Management (IAM) to create users and roles and assign policies that grant limited permissions to end users. The policies define the actions that the user or role can take on your resources through the AWS Marketplace Catalog API.

For example, you can define roles such as engineering, marketing, and pricing. Then, you can add a user in your organization to the engineering role. In that role, they might be granted permissions to initiate a change request to publish a new version of your seller product. However, the engineering role doesn't allow the user to list all change sets.

Note
To sell products on AWS Marketplace, your AWS account must be set up as a seller account. For more details about becoming an AWS Marketplace seller, see Getting started as a seller in the AWS Marketplace Seller Guide.

You can use AWS managed policies, or you can create your own IAM policies to have more granular control than what's available in AWS managed policies. For details about these approaches, see the following topics.

Topics
- Allowing actions with AWS managed policies (p. 10)
- Allowing actions on all resources (p. 11)
- Allowing actions on specific resources (p. 11)
- Allowing actions with specific ChangeType condition key (p. 12)
- Allowing actions with specific aws:ResourceTag condition key (p. 13)
- Managing tags on resources (p. 15)
- Managing tags when requesting changes to resources (p. 17)
- Granting permission to manage tags on resources (p. 18)
- Granting permission to manage tags on resources only when those resources have specific tags (p. 18)
- Granting permission to create entities and change sets only with tags (p. 19)

Allowing actions with AWS managed policies

You can use policies that are managed by AWS to grant permissions to your user or role.

To work with products that you sell on AWS Marketplace, you can use the AWSMarketplaceSellerFullAccess IAM managed policy, which has full access to the AWS...
Marketplace Catalog API in addition to its other permissions. You can grant read-only access for the Catalog API with the AWSMarketplaceSellerProductsReadOnly policy. For more information, see Controlling access to AWS Marketplace Management Portal, Policies and permissions for AWS Marketplace sellers, and AWS managed policies for AWS Marketplace sellers in the AWS Marketplace Seller Guide.

To manage a private marketplace, you can use the AWSPrivateMarketplaceAdminFullAccess IAM managed policy, which has full access to create and edit the private marketplace for your account or AWS organization. For more information, see Controlling access to AWS Marketplace subscriptions, Creating a private marketplace administrator, and AWS managed policies for AWS Marketplace buyers in the AWS Marketplace Buyer Guide.

Alternatively, you can create your own IAM policies to have more granular control than is available in AWS managed policies. Use the following topics to create your own IAM policies.

Allowing actions on all resources

Resources are objects that the actions can act upon. Not every resource type can be specified with every action. Some resource types work with only certain actions. For more information, see Actions, resources, and condition keys for the AWS Marketplace Catalog in the Service Authorization Reference.

There are two resource types in the Catalog API:

- **Entity** – An entity is a seller product in AWS Marketplace or an experience in a private marketplace.
- **ChangeSet** – A change set is returned each time you use Catalog API to make changes to an entity. The change set describes the requested changes and its status. A change set can be canceled if the status is in the PREPARING state.

To allow a user or role the permission to make changes to all entities in an AWS account, you can add the following IAM policy. With this policy, the user or role can use the StartChangeSet action on all resources ("*").

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:StartChangeSet"
      ],
      "Resource": "*"
    }
  ]
}
```

For information about all actions available for the Catalog API, see Actions, resources, and condition keys for AWS Marketplace Catalog in the Service Authorization Reference.

Allowing actions on specific resources

**Note**

Resource-level permissions and condition context keys for the StartChangeSet action are only supported when used with Catalog API. They are not supported when used with the AWS Marketplace Management Portal.

Instead of allowing changes to all resources, you can use resource-level permissions to allow changes to specific resources.
For example, you can allow changes to a specific seller product in the AWS account instead of to all seller products. You do this by specifying the Amazon Resource Name (ARN) of the seller product in the Resource of the IAM policy.

**Note**
To specify granular, resource-level permissions with actions that create new change sets, you need to also include a ChangeSet ARN to the list of resources. The ChangeSet ARN must include the wildcard (/*) to match any new change set ID that's created as shown.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:StartChangeSet"
      ],
      "Resource": [
      ]
    }
  ]
}
```

For another example, you can allow changes to a specific experience in a private marketplace instead of to all experiences. You do this by specifying the ARN of the experience in the Resource of the IAM policy.

**Note**
To specify granular, resource-level permissions with actions that create new change sets, you need to also include a ChangeSet ARN to the list of resources. The ChangeSet ARN must include the wildcard (/*) to match any new change set ID that's created as shown.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:StartChangeSet"
      ],
      "Resource": [
      ]
    }
  ]
}
```

**Allowing actions with specific ChangeType condition key**

**Note**
Resource-level permissions and condition context keys for the `StartChangeSet` action are only supported when used with Catalog API. They are not supported when used with the [AWS Marketplace Management Portal](https://aws.amazon.com/marketplace/).

The Catalog API action `StartChangeSet` has several different change types. You can allow access to only specific change types.
For example, you might only want to allow changes to the metadata of the seller product, such as the product title, and not allow adding new product versions. In this example, the change type UpdateInformation allows changing the metadata of a seller product, including the title. For more information about the different change types, see Working with seller products and Working with a private marketplace in the AWS Marketplace Catalog API Reference.

To limit the action to one or multiple change types, specify the ChangeType in the condition keys. In the following example IAM policy, the condition operator StringEquals specifies that the action is only allowed if the ChangeType matches UpdateInformation. For more information about condition keys, see Condition operators in the AWS Identity and Access Management User Guide.

Note
To specify granular, resource-level permissions with actions that create new change sets, you need to also include a ChangeSet ARN to the list of resources. The ChangeSet ARN must include the wildcard (/*) to match any new change set ID that's created as shown.

```json
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "aws-marketplace:StartChangeSet"
            ],
            "Resource": [
            ],
            "Condition": {
                "StringEquals": {
                    "catalog:ChangeType": "UpdateInformation"
                }
            }
        }
    ]
}
```

### Allowing actions with specific aws:ResourceTag condition key

Note
Resource-level permissions and condition context keys for the StartChangeSet action are only supported when used with Catalog API. They are not supported when used with the AWS Marketplace Management Portal.

You can allow actions on a group of entities without having to keep updating the policy and specifying a possibly growing list of entity ARNs. You can do this with resource tagging. Adding tags to resources allows you to control access to those resources based on their tags. For example, you might want to allow describing a group of seller products without specifying individual ARNs for each seller product.

For example, the following IAM policy allows the DescribeEntity action on any entity resource ("*") that has a tag key of product-team and tag value of team-xyz.

```json
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "aws-marketplace:DescribeEntity"
            ],
            "Resource": [
            ],
            "Condition": {
                "StringEquals": {
                    "catalog:ChangeType": "UpdateInformation"
                }
            }
        }
    ]
}
```
Allowing actions with specific `aws:ResourceTag` condition key

You can also allow describing and canceling change sets that were created with specific tags.

For example, the following IAM policy allows the `DescribeChangeSet` and `CancelChangeSet` actions on any change set resource (`"*"`) that has a tag key of `product-team` and tag value of `team-xyz`.

```json
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "aws-marketplace:DescribeChangeSet",
                "aws-marketplace:CancelChangeSet"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/product-team": "team-xyz"
                }
            }
        }
    ]
}
```

Also, you can allow starting change sets on entities only when those entities have specific tags. For example, you can allow changes to seller products with specific tags.

For example, the following IAM policy allows the `StartChangeSet` action on any entity resource (`"*"`) that has a tag key of `product-team` and tag value of `team-xyz`. In addition, the `TagResource` action is required so that when the change set is created, it's tagged with the same tag key and value.

**Note**

If your policy to allow the `StartChangeSet` action includes a condition to match against specific tags, the same policy must also include the `TagResource` action. This is because the policy condition must match both the tag on the entity and the tag on the newly created change set resulting from the change request. Thus, it requires the user or role to also have the permission to tag the newly created change set. For an example of starting a change set and tagging the change set, see the section called “Example: Adding tags to an entity and change set during creation” (p. 17).

```json
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "aws-marketplace:StartChangeSet",
                "aws-marketplace:TagResource"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/product-team": "team-xyz"
                }
            }
        }
    ]
}
```
Managing tags on resources

You can add, list, and remove tags from existing entities or change sets.

Add tags to resources

To add tags to an entity or change set, use the TagResource API action.

Request

POST /TagResource HTTP/1.1
Content-type: application/json

{
    "Catalog": "AWSMarketplace",
    "ResourceArn": "string",
    "Tags": [
        {
            "Key": "string",
            "Value": "string"
        }...
    ]
}

Request parameters include:

- Catalog (String) – (Required) Must be AWSMarketplace.
- ResourceArn (String) – (Required) ARN of the change set or entity. A change set describes changes you make with Catalog API. An entity can be a seller product in AWS Marketplace or an experience in a private marketplace.
- Tags (Array of objects) – (Required) A list of objects specifying each tag key and value. Number of objects allowed: 1–50.
  - Key (String) – (Required) Name of the tag.
    - Regex pattern – ^([\p{L}\p{Z}\p{N}_.:/=+-@]*)$    
    - Character length – 1–128
  - Value (String) – (Required) Value of the tag.
    - Regex pattern – ^([\p{L}\p{Z}\p{N}_.:/=+-@]*)$  
    - Character length – 0–256

Response

{}

Remove tags from resources

To remove a tag or list of tags from an entity or change set, use the UntagResource API action.
Managing tags on resources

Request

POST /UntagResource HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ResourceArn": "string",
  "TagKeys": [
    "string"
  ]
}

Request parameters include:

- **Catalog (String)** – (Required) Must be AWSMarketplace.
- **ResourceArn (String)** – (Required) ARN of the change set or entity. A change set describes changes you make with Catalog API. An entity can be a **seller product in AWS Marketplace** or an **experience in a private marketplace**.
- **Tags (Array of objects)** – (Required) A list of key names of tags to be removed. Number of strings allowed: 0–256.

Response

{}

**List all tags on a resource**

To list all tags that have been added to and not yet removed from a change set or entity, use the `ListTagsForResource` API action.

Request

POST /ListTagsForResource HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ResourceArn": "string"
}

Request parameters include:

- **Catalog (String)** – (Required) Must be AWSMarketplace.
- **ResourceArn (String)** - (Required) ARN of the change set or entity. A change set describes changes you make with Catalog API. An entity can be a **seller product in AWS Marketplace** or an **experience in a private marketplace**.

Response

{
  "ResourceArn": "string",
  "Tags": [
    {
      "ResourceArn": "string",
      "Tags": [\n        \n      ]
    }
  ]
}
Managing tags when requesting changes to resources

You can add tags when entities or change sets are created.

Example: Adding tags to a change set when creating a change set

The following is an example of a StartChangeSet request that updates the product metadata for a seller product. This request adds a tag to the change set that's created with this request by including the tag in the ChangeSetTags property.

```json
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Identifier": "example1-abcd-1234-5ef6-7890abcdef12",
        "Type": "AmiProduct@1.0"
      },
      "Details": "{"ProductTitle": "My updated title"}"
    }
  ],
  "ChangeSetTags": [
    {
      "Key": "product-team",
      "Value": "team-xyz"
    }
  ]
}
```

For more information about managing seller products, see Working with seller products in the AWS Marketplace Catalog API Reference.

Example: Adding tags to an entity and change set during creation

The following is an example of a StartChangeSet request that creates a private marketplace experience entity. The request adds tags to both the entity resource and change set resource created with this request by including the tags in the EntityTags and ChangeSetTags properties. With these tags, the permission policy of a user or role can be specified to only allow describing or canceling the change set this request creates or only allow creating further change sets on the entity this request creates. For more information, see the section called "Granting permission to create entities and change sets only with tags" (p. 19) section.

```json
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{ /* Omitted for brevity */ }
```
Granting permission to manage tags on resources

To allow a user or role to add, remove, and list tags on all entities or change sets, they need the following IAM policy.

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

Granting permission to manage tags on resources only when those resources have specific tags

You can allow a user or role to add, remove, and list tags on entities or change sets that have specific tags. The following IAM policy allows those actions on any entity resource ("*") that has a tag key of product-team and tag value of team-xyz.

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Resource": "*"
        },{ "Key": "product-team", "Value": "team-xyz" ]
    ]
}
```
Granting permission to create entities and change sets only with tags

Note
Resource-level permissions and condition context keys for the StartChangeSet action are only supported when used with Catalog API. They are not supported when used with the AWS Marketplace Management Portal.

You can enforce tagging when entities or change sets are created. Add the following policy to allow the StartChangeSet and the TagResource actions, with a condition specifying the tag key matches product-team and the tag value matches team-xyz. This policy condition must match both the tag on the newly created entity and the tag on the newly created change set resulting from the creation request. For an example of tagging an entity on creation, see the section called "Example: Adding tags to an entity and change set during creation" (p. 17).

For existing entities, this policy also enforces tagging change sets when requesting changes to those entities. This also requires that the existing entity has this existing tag. This is because the policy condition must match both the tag on the existing entity and the newly created change set resulting from the change request. For an example of adding tags to change requests, see the section called "Example: Adding tags to a change set when creating a change set" (p. 17).

Service quotas
The AWS Marketplace Catalog API has the following quotas.
## Request quotas

<table>
<thead>
<tr>
<th>API operations</th>
<th>Request rate (per AWS account)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ListEntities</td>
<td>10 per second</td>
</tr>
<tr>
<td>DescribeEntity</td>
<td>10 per second</td>
</tr>
<tr>
<td>StartChangeSet</td>
<td>5 per second</td>
</tr>
<tr>
<td>ListChangeSets</td>
<td>5 per second</td>
</tr>
<tr>
<td>DescribeChangeSet</td>
<td>10 per second</td>
</tr>
<tr>
<td>CancelChangeSet</td>
<td>5 per second</td>
</tr>
<tr>
<td>TagResource</td>
<td>5 per second</td>
</tr>
<tr>
<td>UntagResource</td>
<td>5 per second</td>
</tr>
<tr>
<td>ListTagsForResource</td>
<td>5 per second</td>
</tr>
<tr>
<td>PutResourcePolicy</td>
<td>5 per second</td>
</tr>
<tr>
<td>GetResourcePolicy</td>
<td>5 per second</td>
</tr>
<tr>
<td>DeleteResourcePolicy</td>
<td>5 per second</td>
</tr>
</tbody>
</table>

## Account quotas

<table>
<thead>
<tr>
<th>Quota</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum number of open StartChangeSet requests per account</td>
<td>250</td>
</tr>
</tbody>
</table>

## Request history retention quotas

<table>
<thead>
<tr>
<th>Description</th>
<th>Quota</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retention period for change requests. This applies after the end time of each change request.</td>
<td>90 days</td>
</tr>
</tbody>
</table>
Working with seller products

**Note**
Only single-AMI and container-based product types are supported by the AWS Marketplace Catalog API at this time. However, single-AMI with AWS CloudFormation product types are not supported.

You can use the AWS Marketplace Catalog API to manage products that you sell in AWS Marketplace. This chapter assumes that you have access to the API and have completed any seller prerequisites, as described in the [API access control](p. 10) topic.

To understand the basics of using the AWS Marketplace Catalog API, see [AWS Marketplace Catalog API](p. 1).

The following topics describe how to use the Catalog API to perform actions on your single-AMI and container-based products.

**Topics**
- Finding your product ID (server products) (p. 21)
- Change set status and errors (p. 21)
- Working with single-AMI products (p. 26)
- Working with container-based products (p. 38)

Finding your product ID (server products)

You must get the product ID for your product before you can modify it with AWS Marketplace Catalog API. There are two ways to find the product ID for server products:

- Open the AWS Marketplace Management Portal and sign in with your seller account. From the **Products** menu, select **Server products**, then choose the product you are interested in. The product ID is listed in the **Product Summary** section.
- Use the **ListEntities** action with the **EntityType** AmiProduct or ContainerProduct to get a list of products, including their product IDs, via the Catalog API. ListEntities requires that you do not include the version of the entity type (for example, AmiProduct@1.0).

**Note**
The product ID is only available after your product has been published and is visible to at least yourself in AWS Marketplace. When you first create your product, it can take several days to be reviewed and fully created. During this time, it will not have a product ID available.

Change set status and errors

Making changes to seller products in the AWS Marketplace Catalog API involves creating change sets that describe the changes you want to make, and then using the **StartChangeSet** action to start the changes. The changes from the request can take minutes to hours or longer to complete, depending on the request. The response to this request looks like the following.

```json
{
    "ChangeSetId": "example123456789012abcdef",
    "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ChangeSet/example123456789012abcdef"
}
```
The change request is added to a queue and processed, including scanning the files and information to ensure that it meets the AWS Marketplace guidelines for products. Depending on the change requests, this process can take a few minutes to days. You can check the status of the request through the AWS Marketplace Management Portal, or in the Catalog API with the DescribeChangeSet action. For more information about change sets, see Working with change sets (p. 3).

To check the status of your request, use the DescribeChangeSet action.

```
POST /DescribeChangeSet HTTP/1.1
Content-type: application/json
{
    "Catalog": "AWSMarketplace",
    "ChangeSetID": "example123456789012abcdef"
}
```

The result of this call looks like the following (in this case, for adding a new version to a container product).

```
{
    "ChangeSetId": "example123456789012abcdef",
    "ChangeSetName": "Submitted by 123456789012",
    "EndTime": "2020-10-27T22:32:19Z",
    "Status": "SUCCEEDED",
    "ChangeSet": [
        {
            "ChangeType": "AddDeliveryOptions",
            "Entity": {
                "Type": "ContainerProduct@1.0",
                "Identifier": "example-1234-abcd-56ef-abcef12345678@4"
            },
            "Details": "{\"Version\": {\"VersionTitle\": \"1.1\", \"ReleaseNotes\": \"Minor bug fix\"}, \"DeliveryOptions\": [{\"DeliveryOptionTitle\": \"EKSDelivery\", \"Details\": {\"EcrDeliveryOptionDetails\": {\"ContainerImages\": [\"709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame1:1.1\"], \"DeploymentResources\": [{\"Name\": \"HelmDeploymentTemplate\", \"Url\": \"709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame2/mychart1.1\"}], \"CompatibleServices\": [{\"EK5\"}], \"Description\": \"Sample Description\", \"UsageInstructions\": \"helm pull 709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame2:mychart1.1\"}], \"HelmChartDeliveryOption\": {\"Details\": {\"HelmChartUri\": \"709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame1:1.1\"}], \"HelmChartDetails\": {\"CompatibleServices\": [{\"EK5\", \"EK-Anywhere\"}], \"ContainersImages\": [\"709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame1:1.1\"], \"HelmChartDetails\": {\"Description\": \"Helm chart description\", \"UsageInstructions\": \"Usage instructions\", \"QuickLaunchEnabled\": true, \"MarketplaceServiceAccountName\": \"Service account name\", \"ReleaseName\": \"Optional release name\", \"Namespace\": \"Optional Kubernetes namespace\", \"OverrideParameters\": [{\"Key\": \"HelmKeyName1\", \"DefaultValue\": \"$\{AWSMP_LICENSE_SECRET\}"}, \"Metadata\": {\"Label\": \"AWS CloudFormation template field label1\", \"Description\": \"AWS CloudFormation template field description\", \"Obfuscate\": false}, {\"Key\": \"HelmKeyName2\", \"DefaultValue\": \"$\{AWSMP_SERVICE_ACCOUNT\}"}, \"Metadata\": {\"Label\": \"AWS CloudFormation template field label2\", \"Description\": \"AWS CloudFormation template field description\", \"Obfuscate\": false}]}], \"DetailsDocument\": {\"Version\": \n```
"VersionTitle": "1.1",
"ReleaseNotes": "Minor bug fix"
},
"DeliveryOptions": [
{
"DeliveryOptionTitle": "EKSDelivery",
"Details": {
"EcrDeliveryOptionDetails": {
"ContainerImages": [
"709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame1:1.1"
],
"DeploymentResources": [
{
"Name": "HelmDeploymentTemplate",
"Url": "709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame2:mychart1.1"
}
],
"CompatibleServices": [
"EKS"
],
"Description": "Sample Description",
"UsageInstructions": "helm pull 709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame2:mychart1.1"
}
},
"HelmChartDeliveryOption",
"Details": {
"HelmDeliveryOptionDetails": {
"CompatibleServices": [
"EKS",
"EKS-Anywhere"
],
"ContainerImages": [
"709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame1:1.1"
],
"HelmChartUri": "709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame1:helmchart1.1",
"Description": "Helm chart description",
"UsageInstructions": "Usage instructions",
"QuickLaunchEnabled": true,
"MarketplaceServiceAccountName": "Service account name",
"ReleaseName": "Optional release name",
"Namespace": "Optional Kubernetes namespace",
"OverrideParameters": [
{
"Key": "HelmKeyName1",
"DefaultValue": "${AWSMP_LICENSE_SECRET}"
}
]}}
]}}
The Status field shows the current status of the request, in this case, SUCCEEDED.

If there are failures, the result can include two types of errors. For most errors, the error message is included directly. However, errors found while scanning the product for security vulnerabilities instead include a URL to a file that lists all of the errors found, in the ErrorMessage field. Errors found while scanning have the ErrorCode "SCAN_ERROR".

```json
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetName": "Submitted by 123456789012",
  "EndTime": "2020-10-27T22:32:19Z",
  "Status": "FAILED",
  "FailureDescription": "Change set preparation has failed. For details see 'ErrorDetailList'.",
  "ChangeSet": [
    {
      "ChangeType": "AddDeliveryOptions",
      "Entity": {
        "Type": "ContainerProduct@1.0",
        "Identifier": "example-1234-abcd-56ef-abcdef12345678@4"
      },
      "Details": "{\"Version\": {\"VersionTitle\": "1.1\",\"ReleaseNotes\": \"Minor bug fix\"},\"DeliveryOptions\": [{\"DeliveryOptionTitle\": \"EKSDelivery\",\"Details \": {\"EcrDeliveryOptionDetails\": {\"ContainerImages\": [\"709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame1:1.1\"],\"DeploymentResources\": [\"\"],\"Url": "709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame2:mychart1.1\"]},\"CompatibleServices\": [\"EKS\"],\"Description \": "Sample Description\",\"UsageInstructions\": \"helm pull 709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame2:mychart1.1\"}],\"HelmChartDeliveryOption\"},\"Details": {\"HelmDeliveryOptionDetails\": {\"CompatibleServices\": [\"EKS\", \"EKS-Anywhere\"],\"ContainerImages\": [\"709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame1:1.1\"],\"HelmChartUri \": "709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame1:helmchart1.1\",\"Description\": "Helm chart description"},\"UsageInstructions\": \"Usage instructions
```
"QuickLaunchEnabled": true,
"MarketplaceServiceAccountName": "Service account name",
"ReleaseName": "Optional release name",
"Namespaces": "Optional Kubernetes namespace",
"OverrideParameters": [{"Key": "HelmKeyName1", "DefaultValue": "$\{AWSMP\_LICENSE\_SECRET\}\", "Metadata": {"Label": "AWS CloudFormation template field label", "Description": "AWS CloudFormation template field description", "Obfuscate": false}],
"OverrideParameters": [{"Key": "HelmKeyName2", "DefaultValue": "$\{AWSMP\_SERVICE\_ACCOUNT\}\", "Metadata": {"Label": "AWS CloudFormation template field label", "Description": "AWS CloudFormation template field description", "Obfuscate": false}]})},
"DetailsDocument":
{
"Version": {
"VersionTitle": "1.1",
"ReleaseNotes": "Minor bug fix"
},
"DeliveryOptions": [
{
"DeliveryOptionTitle": "EKSDelivery",
"Details": {
"EcrDeliveryOptionDetails": {
"ContainerImages": ["709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame1:1.1"],
"DeploymentResources": [
{
"Name": "HelmDeploymentTemplate",
"Url": "709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame2:mychart1.1"
}
],
"CompatibleServices": [
"EKS"
],
"Description": "Sample Description",
"UsageInstructions": "helm pull 709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame2:mychart1.1"
}
],
"CompatibleServices": [
"EKS",
"EKS-Anywhere"
],
"ContainerImages": [
"709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame1:1.1"
],
"HelmChartUri": "709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame1:helmchart1.1",
"Description": "Helm chart description",
"UsageInstructions": "Usage instructions",
"QuickLaunchEnabled": true,
"MarketplaceServiceAccountName": "Service account name",
}
"ReleaseName": "Optional release name",
"Namespace": "Optional Kubernetes namespace",
"OverrideParameters": [
  {
    "Key": "HelmKeyName1",
    "DefaultValue": "${AWSMP_LICENSE_SECRET}"
  },
  {
    "Key": "HelmKeyName2",
    "DefaultValue": "${AWSMP_SERVICE_ACCOUNT}"
  }
]

"ErrorDetailList": [
  {
    "ErrorCode": "DUPLICATE_VERSION_TITLE",
    "ErrorMessage": "The version title must be different from any other version titles of this product."
  },
  {
    "ErrorCode": "SCAN_ERROR",
    "ErrorMessage": "https://123sample456.cloudfront.net/example-1234-abcd-5678-abcdef12345678/1234abcdef567890"
  }
]

In this example, there is one error directly reported (DUPLICATE_VERSION_TITLE). The other error has a file with error messages (a single SCAN_ERROR can have multiple found errors in the file that is linked).

**Note**
The link returned in the ErrorMessage is valid for 60 days.

## Working with single-AMI products

You can use the AWS Marketplace Catalog API to automate tasks for working with single Amazon Machine Image (AMI-based) products.

With the Catalog API, you can automate updating your existing AMI-based products. You can perform the following actions through the API:

- Update product details
Updating product details for an AMI-based product

As a prerequisite for updating AMI-based products, you must have one or more existing AMI-based products, and you should be familiar with working with the AWS Marketplace Catalog API (p. 1).

Note
For a walk-through showing how to automate updating your AMI-based product, you can also refer to the video, Automating updates to your product listings in AWS Marketplace with Catalog API (5:08).

Updating product details for an AMI-based product

If you already have an AMI-based product in AWS Marketplace, you can use the Catalog API to modify the product details.

You update product details by calling StartChangeSet with the UpdateInformation change type and the details that you want to change, as shown in the following example.

```json
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Identifier": "example1-abcd-1234-5ef6-7890abcdef12",
        "Type": "AmiProduct@1.0"
      },
      "DetailsDocument": {
        "ProductTitle": "My Product Title",
        "ShortDescription": "My product short description.",
        "LongDescription": "My product longer description.",
        "Sku": "123example456",
        "SupportDescription": "Need help? Contact our experts at support@example.com

Your purchase includes 24x7 support.",
        "Categories": [
          "Operating Systems",
          "Network Infrastructure",
          "Application Development"
        ]
      }
    }
  ]
}
```

The following is information about the input fields you provide for adding the UpdateInformation change type:

- **Entity (object)** – Your AMI-based product. The Identifier is your product ID, and the type is always AmiProduct@1.0. For information about using the Identifier, see Identifier (p. 2).
Adding a new version to an AMI-based product

You can use the Catalog API to add a new version to an existing AMI-based product in AWS Marketplace. For more information about adding new AMI versions to your product using the AWS Marketplace Management Portal, see Adding a new version in the AWS Marketplace Seller Guide.

You add a new version in the Catalog API by calling StartChangeSet with the AddDeliveryOptions change type for single-AMI products, as shown in the following example.

**Note**  
For single-AMI products, a version is made up of a single delivery option, which is the AMI that you are making available. In the Catalog API, working with delivery options for single-AMI products modifies versions of your product.
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
    "Catalog": "AWSMarketplace",
    "ChangeSet": [
        {
            "ChangeType": "AddDeliveryOptions",
            "Entity": {
                "Identifier": "example1-abcd-1234-5ef6-7890abcdef12",
                "Type": "AmiProduct@1.0"
            },
            "DetailsDocument": {
                "Version": {
                    "VersionTitle": "*My new title*",
                    "ReleaseNotes": "*My new Release notes*"
                },
                "DeliveryOptions": [
                    {
                        "Details": {
                            "AmiDeliveryOptionDetails": {
                                "AmiSource": {
                                    "AmiId": "ami-123456789abcdef",
                                    "AccessRoleArn": "arn:aws:iam::12345678901:role/AwsMarketplaceAmiIngestion",
                                    "UserName": "ec2-user",
                                    "OperatingSystemName": "AMAZONLINUX",
                                    "OperatingSystemVersion": "Amazon Linux 2 AMI 2.0.20210126.0 x86_64 HVM gp2"
                                },
                                "UsageInstructions": "Easy to use AMI",
                                "RecommendedInstanceType": "m4.xlarge",
                                "SecurityGroups": [
                                    {
                                        "IpProtocol": "tcp",
                                        "FromPort": 443,
                                        "ToPort": 443,
                                        "IpRanges": [
                                            "0.0.0.0/0"
                                        ]
                                    }
                                ]
                            }
                        }
                    }
                ]
            }
        }
    ]
}

The following is information about the input fields you provide for adding the AddDeliveryOptions change type. For more information about these fields, see Adding a new version in the AWS Marketplace Seller Guide.
• **Entity (object)** – Your AMI-based product. The Identifier is your product ID, and the type is always AmiProduct@1.0. For information about using the Identifier, see Identifier (p. 2).

• **DetailsDocument (object)** – Details of the request. It includes all the information about the new version of your AMI-based product.
  - **Version (object)** – Details about the software version you are adding. Made up of a title and release notes.
    - **VersionTitle (string)** – Unique name of the version. Displayed to end users in product details page and configuration pages for the product in AWS Marketplace.
    - **ReleaseNotes (string)** – Notes for buyers to tell them about changes from one version to the next.
  - **DeliveryOptions (array)** – List of DeliveryOption objects, including the details of each:
    - **Details (object)** – Holds the details of an AMI delivery option. Note that this nested details object does not need to be double-escaped.
      - **AmiDeliveryOptionDetails (object)** – The details of one AMI delivery option.
        - **AmiSource (object)** – Details about the AMI to be used for the added version.
          - **AmiId (string)** – ID for the source AMI, located in the AWS Region where the API is being called (currently must always be US East (N. Virginia) because that is the only Region where the Catalog API is available). Must belong to the caller account.
          - **AccessRoleArn (string)** – IAM role Amazon Resource Name (ARN) used by AWS Marketplace to access the provided AMI. For details about creating and using this ARN, see Giving AWS Marketplace access to your AMI in the AWS Marketplace Seller Guide.
          - **UserName (string)** – Login user name to access the operating system (OS) in the AMI. Typically ec2-user for Linux AMIs or Administrator for Windows.
          - **ScanningPort (integer)** – SSH or RDP port used to access the OS. Used for scanning the provided AMI for security vulnerabilities. Defaults to 22.
          - **OperatingSystemName (string)** – Name of the operating system displayed to buyers.
          - **OperatingSystemVersion (string)** – Operating system version string displayed to buyers.
          - **UsageInstructions (string)** – Instructions for using the AMI, or a link to more information about the AMI.
          - **AccessEndpointUrl (object)** – Used to create a path to access the AMI after it is used.
            - **Port (string)** – The port number used to access the service running on the AMI.
            - **Protocol (string)** – The protocol (http or https) used to access the service running on the AMI.
            - **RelativePath (string)** – The path from the web root to access the service running on the AMI (for example /index.html).
          - **RecommendedInstanceType (string)** – The instance type that is recommended to run the service with the AMI and is the default for 1-click installs of your service. For a list of instance types, see Instance types in the Amazon Elastic Compute Cloud User Guide for Linux Instances.
          - **SecurityGroups (array of objects)** – A list of objects representing ingress rules for the automatically created groups for the version.
            - **FromPort (integer)** – The source port.
            - **IpProtocol (string)** – The protocol to use (tcp or udp).
            - **IpRanges (array of strings)** – IP ranges to allow, in CIDR format (in the form xxx.xxx.xxx.xxx/nn, for example, 192.0.2.0/24).
            - **ToPort (integer)** – The destination port.
        - **AmiId (string)** – ID for the source AMI, located in the AWS Region where the API is being called (currently must always be US East (N. Virginia) because that is the only Region where the Catalog API is available). Must belong to the caller account.
      - **Details (object)** – Holds the details of an AMI delivery option. Note that this nested details object does not need to be double-escaped.

A change set is created for your request. The response to this request gives you the ID for the change set and looks like the following.

```json
{
    "ChangeSetId": "example123456789012abcdef",
}
Updating version information for an AMI-based product

The change request is added to a queue and processed, including scanning the files and information to ensure that it meets the AWS Marketplace guidelines for AMI products. This process can take a few minutes to hours. You can check the status of the request through the AWS Marketplace Management Portal, or in the Catalog API with the DescribeChangeSet action. For more information about change sets, see Working with change sets (p. 3). For more information about errors in seller product change sets, see Change set status and errors (p. 21).

When the request is complete, the version is added, and any existing subscribers will receive an email message telling them about the new version. For more information about the process of adding a new version, see Adding a new version in the AWS Marketplace Seller Guide.

Updating version information for an AMI-based product

You can use the Catalog API to update the details of an existing version of your AMI-based product in AWS Marketplace. For more information about updating version information using the AWS Marketplace Management Portal, see Updating version information in the AWS Marketplace Seller Guide. You cannot update the AMI for the version. If you need to update the AMI, create a new version instead.

You add a new version in the Catalog API by calling StartChangeSet with the UpdateDeliveryOptions change type, as shown in the following example.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateDeliveryOptions",
      "Entity": {
        "Identifier": "example1-abcd-1234-5ef6-7890abcdef12",
        "Type": "AmiProduct@1.0"
      },
      "DetailsDocument": {
        "Version": {
          "ReleaseNotes": "*My new Release notes*"
        },
        "DeliveryOptions": [
          {
            "Id": "example1-2222-cccc-2222-cccccccccccc",
            "Details": {
              "AmiDeliveryOptionDetails": {
                "UsageInstructions": "Easy to use AMI"
              }
            }
          }
        ]
      }
    }
  ]
}
```
The following is information about the input fields you provide for adding the UpdateDeliveryOptions change type. For more information about these fields, see Updating version information in the AWS Marketplace Seller Guide.

- **Entity (object)** – Your AMI-based product. The Identifier is your product ID, and the type is always AmiProduct@1.0. For information about using the Identifier, see Identifier (p. 2).
- **DetailsDocument (object)** – Details of the request. It includes any information about the version of your AMI-based product that you would like to update. The included fields are all optional, but you must include at least one field to update.
  - **Version (object)** – Details about the software version.
    - **ReleaseNotes (string)** – Notes for buyers to tell them about changes from one version to the next.
    - **DeliveryOptions (array of objects)** – List of DeliveryOption objects, including the details of each:
      - **Id (string)** – Unique identifier for the DeliveryOption (you can get the unique identifier for the DeliveryOption by calling the DescribeEntity action on the product you are updating).
      - **Details (object)** – Holds the details of an AMI delivery option. Note that this nested details object does not need to be double-escaped.
        - **AmiDeliveryOptionDetails (object)** – The details of one AMI delivery option.
          - **UsageInstructions (string)** – Instructions for using the AMI, or a link to more information about the AMI.
          - **AccessEndpointUrl (object)** – Used to create a path to access the AMI after it is used.
            - **Port (string)** – The port number used to access the service running on the AMI.
            - **Protocol (string)** – The protocol (http or https) used to access the service running on the AMI.
            - **RelativePath (string)** – The path from the web root to access the service running on the AMI (for example /index.html).
          - **RecommendedInstanceType (string)** – The instance type that is recommended to run the service with the AMI and is the default for 1-click installs of your service.
          - **SecurityGroups (array of objects)** – A list of objects representing ingress rules for the automatically created groups for the version:
            - **FromPort (integer)** – The source port.
            - **IpProtocol (string)** – The protocol to use (tcp or idp).
            - **IpRanges (array of strings)** – IP ranges to allow, in CIDR format (in the form xxx.xxx.xxx.xxx/nn, for example, 192.0.2.0/24).
            - **ToPort (integer)** – The destination port.

A change set is created for your request. The response to this request gives you the ID for the change set and looks like the following.

```json
{
  "ChangeSetId": "example123456789012abcdef",
  "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed, including scanning the information to ensure that it meets the AWS Marketplace guidelines for AMI products. This process can take a few minutes to hours. You can check the status of the request through the AWS Marketplace Management Portal, or in the Catalog API with the DescribeChangeSet action. For more information about change sets, see
Restricting a version of an AMI-based product

You can use the Catalog API to restrict a version of your AMI-based product in AWS Marketplace. This prevents new buyers from being able to use that version. There must always be at least one unrestricted version of a product available, so you cannot restrict the last publicly available version for a product. For more information about restricting AMI versions in AWS Marketplace via the AWS Marketplace Management Portal, see Restricting a version in the AWS Marketplace Seller Guide.

You restrict a version in the Catalog API by calling StartChangeSet with the RestrictDeliveryOptions change type, as shown in the following example.

**Note**

All subscribers can use the current version regardless of the restriction status. AWS Marketplace guidelines require that you continue to offer support to existing buyers for 90 days after restricting the version. Your AMI will be marked as deprecated after the version is restricted. For more information, see Deprecate an AMI in the Amazon Elastic Compute Cloud User Guide for Windows Instances.

```json
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "RestrictDeliveryOptions",
      "Entity": {
        "Identifier": "example1-abcd-1234-5ef6-7890abcdef12",
        "Type": "AmiProduct@1.0"
      },
      "DetailsDocument": {
        "DeliveryOptionIds": [
          "example1-2222-cccc-2222-cccccccccccc"
        ]
      }
    }
  ]
}
```

The following is information about the input fields you provide for adding the RestrictDeliveryOptions change type:

- **Entity (object)** – Your AMI-based product. The Identifier is your product ID, and the type is always AmiProduct@1.0. For information about using the Identifier, see Identifier (p. 2).
- **DetailsDocument (object)** – Details of the request. It includes IDs for the versions of your AMI-based product that you would like to restrict.
  - **DeliveryOptionIds (array of objects)** – List of DeliveryOption IDs for the versions that you want to restrict. You can get the unique identifier for the DeliveryOption by calling the DescribeEntity action on the version you are restricting.

A change set is created for your request. The response to this request gives you the ID for the change set and looks like the following.

---

Working with change sets (p. 3). For more information about errors in seller product change sets, see Change set status and errors (p. 21).

Restricting a version of an AMI-based product

You can use the Catalog API to restrict a version of your AMI-based product in AWS Marketplace. This prevents new buyers from being able to use that version. There must always be at least one unrestricted version of a product available, so you cannot restrict the last publicly available version for a product. For more information about restricting AMI versions in AWS Marketplace via the AWS Marketplace Management Portal, see Restricting a version in the AWS Marketplace Seller Guide.

You restrict a version in the Catalog API by calling StartChangeSet with the RestrictDeliveryOptions change type, as shown in the following example.

**Note**

All subscribers can use the current version regardless of the restriction status. AWS Marketplace guidelines require that you continue to offer support to existing buyers for 90 days after restricting the version. Your AMI will be marked as deprecated after the version is restricted. For more information, see Deprecate an AMI in the Amazon Elastic Compute Cloud User Guide for Windows Instances.

```json
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "RestrictDeliveryOptions",
      "Entity": {
        "Identifier": "example1-abcd-1234-5ef6-7890abcdef12",
        "Type": "AmiProduct@1.0"
      },
      "DetailsDocument": {
        "DeliveryOptionIds": [
          "example1-2222-cccc-2222-cccccccccccc"
        ]
      }
    }
  ]
}
```

The following is information about the input fields you provide for adding the RestrictDeliveryOptions change type:

- **Entity (object)** – Your AMI-based product. The Identifier is your product ID, and the type is always AmiProduct@1.0. For information about using the Identifier, see Identifier (p. 2).
- **DetailsDocument (object)** – Details of the request. It includes IDs for the versions of your AMI-based product that you would like to restrict.
  - **DeliveryOptionIds (array of objects)** – List of DeliveryOption IDs for the versions that you want to restrict. You can get the unique identifier for the DeliveryOption by calling the DescribeEntity action on the version you are restricting.

A change set is created for your request. The response to this request gives you the ID for the change set and looks like the following.
The change request is added to a queue and processed. This process can take a few minutes to hours. You can check the status of the request through the AWS Marketplace Management Portal, or in the Catalog API with the DescribeChangeSet action. For more information about change sets, see Working with change sets (p. 3). For more information about errors in seller product change sets, see Change set status and errors (p. 21).

Errors in the AMI products API

The following errors are specific to the AMI product actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For more details about using DescribeChangeSet to get the status of a change request, see Working with change sets (p. 3).

<table>
<thead>
<tr>
<th>Change type</th>
<th>Error code</th>
<th>Error message</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_PRODUCT</td>
<td>Use an existing limited or public product.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>DUPLICATE_VERSION_TITLE</td>
<td>The version title must be different from any other version titles of this product.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_VERSION_TITLE</td>
<td>Remove spaces before the trademark symbol.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_VERSION_TITLE</td>
<td>Remove unsupported characters: [x, y, z]</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_VERSION_TITLE</td>
<td>Remove spaces from the beginning of the version title.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_VERSION_TITLE</td>
<td>Provide version title with fewer than [x] characters.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_RELEASE_NOTES</td>
<td>Remove spaces before the trademark symbol.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_RELEASE_NOTES</td>
<td>Remove unsupported characters: [x, y, z]</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_RELEASE_NOTES</td>
<td>Remove spaces from the beginning of release notes.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_RELEASE_NOTES</td>
<td>Provide release notes with fewer than (x) characters.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_USAGE_INSTRUCTIONS</td>
<td>Remove spaces before the trademark symbol.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_USAGE_INSTRUCTIONS</td>
<td>Remove unsupported characters: [x, y, z]</td>
</tr>
<tr>
<td>Change type</td>
<td>Error code</td>
<td>Error message</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------------------------</td>
<td>----------------------------------------------------</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_USAGE_INSTRUCTIONS</td>
<td>Remove spaces from the beginning of release notes.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_USAGE_INSTRUCTIONS</td>
<td>Provide usage instructions with fewer than (x) characters.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>RECOMMENDED_INSTANCE_TYPE</td>
<td>Provide a valid instance type.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_SECURITY_GROUP</td>
<td>Security group ports must be between 1 and [max].</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_SECURITY_GROUP</td>
<td>Provide a value for CIDR IP ranges.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_SECURITY_GROUP</td>
<td>Provide security group start port that is not greater than end port.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_SECURITY_GROUP_PROTOCOL</td>
<td>Security group protocol must either be 'tcp' or 'udp'.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_CIDR_IP</td>
<td>Provide standard CIDR IP range in form '0.0.0.0/0'.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_ACCESS_ENDPOINT_PORT</td>
<td>Provide endpoint port less than [x].</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_ACCESS_ENDPOINT_PORT</td>
<td>Provide endpoint port between 1 and [max].</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_ACCESS_ENDPOINT_PORT</td>
<td>Provide endpoint port.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_ACCESS_ENDPOINT_RELATIVE_PATH</td>
<td>Remove spaces in the relative path.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_ACCESS_ENDPOINT_RELATIVE_PATH</td>
<td>Remove preceding '/' from relative path.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INCOMPATIBLE_OPERATING_SYSTEM</td>
<td>Provide operating system name and version that is compatible with instance types: [x]</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INCOMPATIBLE_OPERATING_SYSTEM_NAME</td>
<td>Provide name with fewer than (x) characters.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INCOMPATIBLE_OPERATING_SYSTEM_VERSION</td>
<td>Provide version with fewer than (x) characters.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_SCANNING_PORT</td>
<td>Provide scanning port between 1 and [max].</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_AMI_ID</td>
<td>Provide valid AMI ID.</td>
</tr>
<tr>
<td>Change type</td>
<td>Error code</td>
<td>Error message</td>
</tr>
<tr>
<td>----------------------</td>
<td>-------------------------------------------------</td>
<td>-------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>EXISTING_AMI_PRODUCT_CODE</td>
<td>Remove product code attached to image X.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_AMI_ARCHITECTURE</td>
<td>Provide new AMI with architecture [x].</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_AMI_VIRTUALIZATION_TYPE</td>
<td>Provide new AMI with virtualization type [x].</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_AMI_VIRTUALIZATION_TYPE</td>
<td>Provide expected [z] volume on image [x].</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INCOMPATIBLE_AMI</td>
<td>Provide new AMI as architecture [x] on [y] is not supported by following instance types: [z]</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INCOMPATIBLE_AMI</td>
<td>Provide new AMI as virtualization type [x] on [y] is not supported by following instance types: [z]</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INCOMPATIBLE_AMI</td>
<td>Enable ENA support for image x because following instance types require ENA support: [y]</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>ASSET_NOT_FOUND</td>
<td>Check if [ami-id] exists in us-east-1 Region of [account-id] AWS account and the AccessARN provided [ARN] has permissions to share this AMI with AWS Marketplace.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>ASSET_ACCESS_EXCEPTION</td>
<td>Unable to copy AMI [x] into AWS Marketplace account.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>SCAN_ERROR</td>
<td>Fix security vulnerability [y] on image [x].</td>
</tr>
<tr>
<td>RestrictDeliveryOptions</td>
<td>INVALID_PRODUCT</td>
<td>Use an existing public product.</td>
</tr>
<tr>
<td>RestrictDeliveryOptions</td>
<td>MISSING_DELIVERY_OPTION_IDS</td>
<td>Provide at least one delivery option ID.</td>
</tr>
<tr>
<td>RestrictDeliveryOptions</td>
<td>INVALID_DELIVERY_OPTION_IDS</td>
<td>Provide delivery option IDs that can be found in the product. IDs not found: [x]</td>
</tr>
<tr>
<td>RestrictDeliveryOptions</td>
<td>INVALID_DELIVERY_OPTION_IDS</td>
<td>Provide delivery option IDs that are in a public state. IDs not in public state: [x]</td>
</tr>
<tr>
<td>RestrictDeliveryOptions</td>
<td>ALL_DELIVERY_OPTIONS_RESTRICTED</td>
<td>Provide fewer delivery options to restrict as at least one must remain in public state.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_PRODUCT</td>
<td>Use an existing limited or public product.</td>
</tr>
<tr>
<td>Change type</td>
<td>Error code</td>
<td>Error message</td>
</tr>
<tr>
<td>------------------------</td>
<td>---------------------------------------------</td>
<td>-------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>MISSING_DELIVERY_OPTION_IDS</td>
<td>Provide at least one delivery option ID.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_DELIVERY_OPTION_IDS</td>
<td>Provide delivery option IDs that can be found in the product. IDs not found: [x]</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_DELIVERY_OPTIONS</td>
<td>Provide delivery option IDs that belong to the same version.</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>MISSING_UPDATES</td>
<td>Nothing to update. Provide updated information for product.</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>MISSING_LOGO_URL</td>
<td>Provide URL for logo stored in S3.</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>LOGO_COPY_FAILURE</td>
<td>There was an issue copying the logo from S3. Provide a new URL for logo stored in S3.</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_CATEGORY_NAMES</td>
<td>Provide valid category names supported by AWS Marketplace.</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_PRODUCT_TITLE</td>
<td>Remove spaces before trademark symbol.</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_PRODUCT_TITLE</td>
<td>Remove unsupported characters [x, y, z].</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_SHORT_DESCRIPTION</td>
<td>Remove spaces before trademark symbol.</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_SHORT_DESCRIPTION</td>
<td>Remove unsupported characters [x, y, z].</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_LONG_DESCRIPTION</td>
<td>Remove spaces before trademark symbol.</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_LONG_DESCRIPTION</td>
<td>Remove unsupported characters [x, y, z].</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_SUPPORT_DESCRIPTION</td>
<td>Remove spaces before trademark symbol.</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_SUPPORT_DESCRIPTION</td>
<td>Remove unsupported characters [x, y, z].</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_HIGHLIGHTS</td>
<td>Remove spaces before trademark symbol.</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_HIGHLIGHTS</td>
<td>Remove unsupported characters [x, y, z].</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_ADDITIONAL_RESOURCES</td>
<td>Remove spaces before trademark symbol.</td>
</tr>
<tr>
<td>Change type</td>
<td>Error code</td>
<td>Error message</td>
</tr>
<tr>
<td>---------------------</td>
<td>------------------------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_ADDITIONAL_RESOURCES</td>
<td>Remove unsupported characters ([x, y, z]).</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_PRODUCT_CATEGORIES</td>
<td>Provide between 1 and 3 product categories.</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_IMAGE_PROPERTIES</td>
<td>Provide an image that meets the product logo requirements. (link)</td>
</tr>
</tbody>
</table>

**Working with container-based products**

You can use the AWS Marketplace Catalog API to automate tasks for working with container-based products.

With the Catalog API, you can automate updating your existing container-based products. You can perform the following actions through the API:

- Update product information
- Add a new version
- Add repositories for your container images and other resources
- Update a version
- Restrict a version

As a prerequisite for updating container-based products, you must have one or more existing container-based products, and you should be familiar with working with the AWS Marketplace Catalog API (p. 1).

**Note**

For details about creating a container-based product through the AWS Marketplace Management Portal, see [Getting started with container products](#) in the AWS Marketplace Seller Guide.

**Updating product information for a container-based product**

If you already have a container-based product in AWS Marketplace, you can use the Catalog API to modify the product information.

You update product information by calling `StartChangeSet` with the `UpdateInformation` change type and the details that you want to change, as shown in the following example.

```plaintext
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateInformation",
      "Entity": {
        "Identifier": "example1-abcd-1234-5ef6-7890abcdef12",
        "Type": "ContainerProduct@1.0"
      }
    }
  ]
}
```
AWS Marketplace Catalog API Reference

Updating product information for a container-based product

The following is information about the input fields you provide for adding the UpdateInformation change type:

- **Entity (object)** – The container-based product that you want to update. The Identifier is your product ID, and the type is always ContainerProduct@1.0. For information about using the Identifier, see Identifier (p. 2).
- **DetailsDocument (object)** – Details of the request. It includes all the information you want to update for your product. Each field is optional, but you must include at least one change to update.
  - **ProductTitle (string)** – Name of the product to be displayed to buyers.
  - **ShortDescription (string)** – Description of key aspects of the product to be displayed to buyers. Typically 2–3 sentences.
  - **LongDescription (string)** – Longer description of your product to be displayed to buyers. Typically 1–3 paragraphs.
  - **Sku (string)** – A free-form string for you to define as a reference for your own use.
  - **LogoUrl (string)** – A URL to an image in a publicly accessible Amazon S3 bucket. For more details about image formats, see Company and product logo requirements in the AWS Marketplace Seller Guide.
  - **VideoUrls (array of strings)** – A list of URLs to publicly available, externally hosted videos to be provided as a reference to buyers in your product information.
  - **Highlights (array of strings)** – A list of short callouts for key product features.
  - **AdditionalResources (array of objects)** – List of references to additional resources to learn about your product. Each reference is made up of a text name and a URL:
    - **Text (string)** – The name or title of the resource.
    - **Url (string)** – A URL to a resource that is helpful for a buyer to understand your product.
  - **SupportDescription (string)** – Details about your support offerings for your product.
  - **Categories (array of strings)** – A list of AWS Marketplace defined product categories that describe your product.
  - **SearchKeywords (array of strings)** – A list of additional keywords for your product for the search experience. Seller name, product name, and product categories are automatically included in search keywords and do not need to be repeated here.

A change set is created for your request. The response to this request gives you the ID for the change set and looks like the following:

```json
{
}
```
Adding a new version to a container-based product

If you already have a container-based product in AWS Marketplace, you can use the Catalog API to add a new version. This requires that you have already created repositories in AWS Marketplace for each container image or artifact that is part of your product, and that you can copy them from your local Docker and Helm files.

**Note**
For details about creating a container-based product, see [Getting started with container products](#).
For details about adding a new version, including creating repositories and building Docker and Helm files into those repositories, by using the AWS Marketplace Management Portal, see [Add a new version of your product](#) in the [AWS Marketplace Seller Guide](#).
If you have not already created new repositories, you can create them using the Catalog API, see [Creating repositories and resources for a container-based product](#).

You add a new version by calling `StartChangeSet` with the `AddDeliveryOptions` change type, as shown in the following example.

**Note**
A version of a container-based product is made up of one or more delivery options. For example, you might have two delivery options, one that works with a noSQL database, and another that works with MySQL, so that your users can choose how they want to work with your product. You create the version of your product and add multiple delivery options in a single request with `AddDeliveryOptions`.

```json
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "AddDeliveryOptions",
      "Entity": {
        "Identifier": "example1-abcd-1234-5ef6-7890abcdef12",
        "Type": "ContainerProduct@1.0"
      },
      "DetailsDocument": {
        "Version": {
          "VersionTitle": "1.1",
          "ReleaseNotes": "Minor bug fix"
        }
      },
      "DeliveryOptions": 
    }
  }
}
```

The change request is added to a queue and processed, including scanning the files and information to ensure that it meets the AWS Marketplace guidelines for products. This process can take a few minutes to hours. You can check the status of the request through the AWS Marketplace Management Portal, or in the Catalog API with the `DescribeChangeSet` action. For more information about change sets, see [Working with change sets](#). For more information about errors in seller product change sets, see [Change set status and errors](#).

---

“ChangeSetId”: "example123456789012abcdef",
}
[{
   "DeliveryOptionTitle": "EKSDelivery",
   "Details": {
      "EcrDeliveryOptionDetails": {
         "ContainerImages": [
            "709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame1:1.1"
         ],
         "DeploymentResources": [
            {
               "Name": "HelmDeploymentTemplate",
               "Url": "709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame2:mychart1.1"
            }
         ],
         "CompatibleServices": [
            "EKS"
         ],
         "Description": "Sample Description",
         "UsageInstructions": "helm pull 709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame2:mychart1.1"
      }
   }
},
{
   "DeliveryOptionTitle": "HelmChartDeliveryOption",
   "Details": {
      "HelmDeliveryOptionDetails": {
         "CompatibleServices": [
            "EKS",
            "EKS-Anywhere"
         ],
         "ContainerImages": [
            "709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame1:1.1"
         ],
         "HelmChartUri": "709825985650.dkr.ecr.us-east-1.amazonaws.com/sellername/reponame1:helmchart1.1",
         "Description": "Helm chart description",
         "UsageInstructions": "Usage instructions",
         "QuickLaunchEnabled": true,
         "MarketplaceServiceAccountName": "Service account name",
         "ReleaseName": "Optional release name",
         "Namespace": "Optional Kubernetes namespace",
         "OverrideParameters": [
            {
               "Key": "HelmKeyName1",
               "DefaultValue": "${AWSMP_LICENSE_SECRET}"
            },
            {
               "Key": "HelmKeyName2",
               "DefaultValue": "false"
            }
         ]
      }
   }
}]

AWS Marketplace Catalog API Reference
Adding a new version to a container-based product
Adding a new version to a container-based product

```
"DefaultValue": "${AWSMP_SERVICE_ACCOUNT}",
"Metadata": {
  "Label": "AWS CloudFormation template field label",
  "Description": "AWS CloudFormation template field description",
  "Obfuscate": false
}
}
```

The following is information about the input fields you must provide for adding the AddDeliveryOptions change type for container-based products:

- **Entity (object)** – Your container-based product. The Identifier is your product ID, and the type is always ContainerProduct@1.0. For information about using the Identifier, see Identifier (p. 2).
- **DetailsDocument (object)** – Details of the request. It includes all the information about the version that you are adding. This field is a string field.
- **Version (object)** – Details about the version that you are adding to your product.
  - **VersionTitle (string)** – The title of the version that you are creating. Typically this is a description of the version, like Version 1.1 or simply 1.1. Your buyers will be able to choose the version to deploy from a list of version titles.
  - **ReleaseNotes (string)** – The detailed notes about this version. Must be less than 30,000 characters.
- **DeliveryOptions (array of objects)** – An array of delivery options, where each is a method of delivery for your product version. For example, if you have one delivery option for Amazon Elastic Container Service (Amazon ECS) and another for Amazon Elastic Kubernetes Service (Amazon EKS), you will need to have two delivery options.
  - **DeliveryOptionTitle (string)** – A short description that helps your buyer to choose between your delivery options.
  - **Details (string)** – The resources used for this delivery option. This is a details field within the details field. You do not need to doubly escape characters in this field.
  - **ContainerImages (array of strings)** – An array of container image URLs used by this version. The path will be the repository that you have uploaded the image to, with the tag for the image used by this version. The list must include all needed images, even images that have not changed from previous versions. See the next section for information about creating repositories using the Catalog API.
  - **EcrDeliveryOptionDetails** – **DeploymentResources (array of objects)** – An array of other resources needed for the version, such as Helm charts. Each resource includes a Name to describe it, and a URL that points at the resource.
  - **HelmDeliveryOptionDetails** – **HelmChartUri (string)** – The URL to the Helm chart hosted in Amazon ECR that the buyer will install to launch the software.
  - **HelmDeliveryOptionDetails** – **QuickLaunchEnabled (boolean)** – A boolean to determine if buyers can use QuickLaunch to launch the software. For more information about QuickLaunch, see QuickLaunch in AWS Marketplace.
  - **HelmDeliveryOptionDetails** – **MarketplaceServiceAccountName (string)** – Optional – The name of the Kubernetes service account. The service account will be used to connect to AWS Identity and Access Management (IAM) for permissions to call AWS services.
Creating repositories and resources for a container-based product

To create a new version of a container-based product, you must have the resources for the version available in AWS Marketplace repositories. You create the repositories and then push (upload) the Docker
Creating repositories and resources for a container-based product

To learn how to create the repositories through the AWS Marketplace Management Portal, see Add a new version of your product in the AWS Marketplace Seller Guide.

To create new repositories with the Catalog API, call StartChangeSet with the AddRepositories change type, as shown in the following example.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "AddRepositories",
      "Entity": {
        "Identifier": "example1-abcd-1234-5ef6-7890abcdef12",
        "Type": "ContainerProduct@1.0"
      },
      "DetailsDocument": {
        "Repositories": [
          {
            "RepositoryName": "new-repo-1",
            "RepositoryType": "ECR"
          },
          {
            "RepositoryName": "new-repo-2",
            "RepositoryType": "ECR"
          }
        ]
      }
    }
  ]
}
```

The following is information about the input fields you provide for adding the AddRepositories change type. For more information about creating repositories, see Adding a new version in the AWS Marketplace Seller Guide.

- **Entity (structure)** – The container-based product you are creating repositories for. The Identifier is your product ID, and the type is always ContainerProduct@1.0. For information about using the Identifier, see Identifier (p. 2).
- **DetailsDocument (object)** – Details of the request. It includes the information about the repositories that you want to create. The included fields are all required.
  - **Repositories (array of structures)** – A list of repository objects. Each repository object includes a name and type.
    - **RepositoryName (string)** – The name of the repository to create.
    - **RepositoryType (string)** – The type of the repository to create. The only allowed value is ECR.

**Note**
You can only have 50 repositories per product, although you can add multiple resources (and versions of resources) to a single repository by giving them different tags when you push them.

After you have created one or more repositories for your resources, you add your resources to the repositories. For general information about how to push resources to repositories, see Pushing an image.
Updating version information for a container-based product

You can use the Catalog API to update the details of an existing version of your container-based product in AWS Marketplace.

Note
When a product is publicly available, you cannot update the version title, container images, delivery option title, or deployment resources for the version. If you need to update these aspects of a product, create a new version instead.

You update an existing version of your container-based product in the Catalog API by calling the `StartChangeSet` with the `UpdateDeliveryOptions` change type, as shown in the following example. This updates the detail information for the delivery options that you specify, as well as the associated version. You must include at least one delivery option.

```json
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateDeliveryOptions",
      "Entity": {
        "Identifier": "example1-abcd-1234-5ef6-7890abcdef12",
        "Type": "ContainerProduct@1.0"
      },
      "DetailsDocument": {
        "Version": {
          "ReleaseNotes": "New release notes",
          "VersionTitle": "Version 1.2"
        },
        "DeliveryOptions": [
          {
            "Id": "example4-2222-cccc-2222-cccccccccccc",
            "Details": {
              "EcrDeliveryOptionDetails": {
                "DeliveryOptionTitle": "New Delivery Option Title",
                "Description": "New description",
                "UsageInstructions": "New usage instructions",
                "CompatibleServices": [
                  "EKS"
                ],
              }
            }
          },
          {
            "Id": "example5-2222-cccc-2222-cccccccccccc",
            "Details":
          }
        ]
      }
    }
  ]
}
```
The following is information about the input fields you provide for adding the UpdateDeliveryOptions change type for container-based products. For more information about these fields, see Adding a new version in the AWS Marketplace Seller Guide.

- **Entity (object)** – Your container-based product. The Identifier is your product ID, and the type is always ContainerProduct@1.0. For information about using the Identifier, see **Identifier (p. 2)**.

- **DetailsDocument (object)** – Details of the request. It includes any information about the version of your container-based product that you would like to update. The included fields are all optional, but you must include at least one field to update.

- **Version (object)** – Details about the software version.
  - **VersionTitle (string)** – The title of the version that you are creating. Typically this is a description of the version, such as **Version 1.1** or simply **1.1**. Your buyers will be able to choose the version to deploy from a list of all version titles.

    This property can't be updated if the product is already published publicly.

- **ReleaseNotes (string)** – Notes for buyers to tell them about changes from one version to the next.

- **DeliveryOptions (list of objects)** – List of DeliveryOption objects, including the details of each:

  - **Id (string)** – Unique identifier for the DeliveryOption (you can get the unique identifier for the DeliveryOption by calling the DescribeEntity action on the product you are updating).
• **Details (string)** – Holds the details of a delivery option. Note that this nested details object does not need to be double-escaped.

• **EcrDeliveryOptionDetails (object)** – The details of the container image delivery option.
  • **DeliveryOptionTitle (string)** – A short description that allows your buyer to choose between your delivery options.

This property can't be updated if the product is already published publicly.

• **ContainerImages (array of strings)** – An array of container image URLs used by this version. The path will be the repository that you have uploaded the image to, with the tag for the image used by this version. If this field is included, the list must include all needed images, even images that are not changing.

This property can't be updated if the product is already published publicly.

• **DeploymentResources (array of objects)** – An array of other deployment resources needed for the version, such as links to Helm charts or other documentation. Each resource includes a name to describe it and a URL that points at the resource. On the launch page for your version, this displays as a list of links.

This property can't be updated if the product is already published publicly.

• **Name (string)** – The text of the hyperlink that is shown to the buyer.

• **Url (string)** – The URL of the hyperlink shown to the buyer.

• **CompatibleServices (array of strings)** – A list of services that the release is compatible with. Valid options are ECS and EKS.

• **Description (string)** – A longer description of the delivery option to give details to your buyer. You can also include a link to more instructions hosted elsewhere.

• **UsageInstructions (string)** – Provide instructions on how to deploy and use your product. You can also add a link to usage instructions hosted elsewhere. Can be up to 4,000 characters.

• **Id (string)** – Unique identifier for the DeliveryOption (you can get the unique identifier for the DeliveryOption by calling the DescribeEntity action on the product you are updating).

• **Details (string)** – Holds the details of a delivery option. Note that this nested details object does not need to be double-escaped.

• **HelmDeliveryOptionDetails (object)** – The details of the Helm chart delivery option.
  • **DeliveryOptionTitle (string)** – A short description that allows your buyer to choose between your delivery options.

This property can't be updated if the product is already published publicly.

• **ContainerImages (array of strings)** – An array of container image URLs used by this version. The path will be the repository that you have uploaded the image to, with the tag for the image used by this version. The list must include all needed images, even images that have not changed from previous versions. See the next section for information about creating repositories using the Catalog API.

• **HelmChartUri (string)** – The URL to the Helm chart hosted in Amazon ECR that the buyer will install to launch the software.

• **CompatibleServices (array of strings)** – An array of services that the release is compatible with. Valid options are ECS and EKS.

• **Description (string)** – A longer description of the delivery option to give details to your buyer. You can also include a link to more instructions provided elsewhere.

• **UsageInstructions (string)** – Provide instructions about the usage for this delivery option. Can be up to 4,000 characters.

• **MarketplaceServiceAccountName (string)** – The name of the Kubernetes service account. The service account will be used to connect to AWS Identity and Access Management for permissions to call AWS services.
Restricting a version of a container-based product

You can use the Catalog API to restrict a version of your container-based product in AWS Marketplace. This prevents new buyers from being able to use that version. There must be at least one publicly available version in a product. You cannot restrict the only remaining publicly available version for a product.

You restrict a version in the Catalog API by calling StartChangeSet with the RestrictDeliveryOptions change type, as shown in the following example.

**Note**
Restricting one or more, but not all, delivery options from a version will remove those options from being available to your buyers. Restricting all delivery options for a version will remove that version from the AWS Marketplace catalog.

Restricted versions are still available for existing customers.

```plaintext
POST /StartChangeSet HTTP/1.1
```
Errors in the container products API

The following information about the input fields you provide for adding the RestrictDeliveryOptions change type:

- **Entity (object)** – Your container-based product. The Identifier is your product ID, and the type is always ContainerProduct@1.0. For information about using the Identifier, see Identifier (p. 2).
- **DetailsDocument (object)** – Details of the request. It includes IDs for the delivery options of your container-based product that you would like to restrict.
- **DeliveryOptionIds (array of strings)** – List of DeliveryOption IDs for the versions that you want to restrict. You can get the unique identifier for the DeliveryOption by calling the DescribeEntity action on the product you are restricting.

A change set is created for your request. The response to this request gives you the ID for the change set and looks like the following.

```json
{
   "ChangeSetId": "example123456789012abcdef",
   "ChangeSetArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ChangeSet/example123456789012abcdef"
}
```

The change request is added to a queue and processed. This process can take a few minutes to hours. You can check the status of the request through the AWS Marketplace Management Portal, or in the Catalog API with the DescribeChangeSet action. For more information about change sets, see Working with change sets (p. 3). For more information about errors in seller product change sets, see Change set status and errors (p. 21).

### Errors in the container products API

The following errors are specific to the container product actions in the AWS Marketplace Catalog API. These errors are returned when you call DescribeChangeSet after a change set is processing. For more details about using DescribeChangeSet to get the status of a change request, see Working with change sets (p. 3).
<table>
<thead>
<tr>
<th>Change type</th>
<th>Error code</th>
<th>Error message</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddDeliveryOptions</td>
<td>INCOMPATIBLE_PRODUCT_STATUS</td>
<td>Use an existing limited or public product.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INCOMPATIBLE_SERVICES</td>
<td>Provide a valid list of compatible services.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>NO_SERVICE_SPECIFIED</td>
<td>Provide at least 1 compatible service.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>DUPLICATE_COMPATIBLE_AWS_SERVICES</td>
<td>Provide a unique list of compatible services.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_VERSION_TITLE</td>
<td>Remove spaces before the trademark symbol.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_VERSION_TITLE</td>
<td>Remove the following unsupported characters: [x, y, z]</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_VERSION_TITLE</td>
<td>Remove spaces from the beginning of the version title.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_VERSION_TITLE</td>
<td>Provide version title with fewer than [x] characters.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>DUPLICATE_VERSION_TITLE</td>
<td>The version title must be different from any other version titles of this product.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_RELEASE_NOTES</td>
<td>Remove spaces before the trademark symbol.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_RELEASE_NOTES</td>
<td>Remove unsupported characters: [x, y, z]</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_RELEASE_NOTES</td>
<td>Remove spaces from the beginning of release notes.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_RELEASE_NOTES</td>
<td>Provide release notes with fewer than (x) characters.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_USAGE_INSTRUCTIONS</td>
<td>Remove spaces before the trademark symbol.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_USAGE_INSTRUCTIONS</td>
<td>Remove unsupported characters: [x, y, z]</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_USAGE_INSTRUCTIONS</td>
<td>Provide usage instructions with fewer than (x) characters.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>NO_LICENSE_SECRET_KEYS</td>
<td>Provide usage instructions.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>MISSING_CONTAINER_IMAGES</td>
<td>Provide at least 1 container image.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>NO_LICENSE_SECRET_KEYS</td>
<td>For Amazon EKS Anywhere products, provide 1 override parameter for license secret. Needs DefaultValue of</td>
</tr>
<tr>
<td>Change type</td>
<td>Error code</td>
<td>Error message</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>TOO_MANY_CONTAINER_IMAGES</td>
<td>Provide fewer than 50 container images.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>DUPLICATE_CONTAINER_IMAGES</td>
<td>Provide a unique list of container images.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_CONTAINER_IMAGES</td>
<td>Provide a valid URI for the container image.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_CONTAINER_IMAGE_URI</td>
<td>Provide a valid URI for the container image.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_CONTAINER_IMAGE_TAG</td>
<td>Avoid using 'latest' tag.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>DUPLICATE_DELIVERY_OPTION_TITLE</td>
<td>Provide unique delivery option title.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_DELIVERY_OPTION_TITLE</td>
<td>Delivery option title already exists, retry with a different title.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_FULFILLMENT_OPTION_TITLE</td>
<td>Provide delivery option title with fewer than (x) characters.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>DUPLICATE_DELIVERY_OPTION_TITLE</td>
<td>Provide unique delivery option title.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>NO_SERVICE_ACCOUNT_CONFIGURATION</td>
<td>For paid products, provide 1 override parameter for service account configuration. Needs DefaultValue of &quot;${AWSMP_SERVICE_ACCOUNT}&quot;, see example in section.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>EMPTY_RESOURCE_NAME</td>
<td>Provide resource name.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>EMPTY_RESOURCE_URL</td>
<td>Provide resource URL.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_RESOURCE_NAME</td>
<td>Provide resource name with fewer than 256 characters.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_RESOURCE_URL</td>
<td>Provide resource URL with fewer than 256 characters.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_SHORT_DESCRIPTION</td>
<td>Provide a short description with fewer than 1,000 characters.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_SHORT_DESCRIPTION</td>
<td>Provide short description.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>SCAN_ERROR</td>
<td>Fix security vulnerability &quot;[y]&quot; on Image &quot;'[x]'&quot;.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>IMAGE_NOT_FOUND</td>
<td>Provide a valid public image URI.</td>
</tr>
<tr>
<td>Change type</td>
<td>Error code</td>
<td>Error message</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>-------------------------------------</td>
<td>----------------------------------------------------</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>INVALID_ARN</td>
<td>Provide a valid ARN for image access.</td>
</tr>
<tr>
<td>AddDeliveryOptions</td>
<td>IMAGE_INACCESSIBLE</td>
<td>Provide a valid ARN for image access.</td>
</tr>
<tr>
<td>RestrictDeliveryOptions</td>
<td>INCOMPATIBLE_PRODUCT_STATUS</td>
<td>Use a public product.</td>
</tr>
<tr>
<td>RestrictDeliveryOptions</td>
<td>MISSING_DELIVERY_OPTION_IDS</td>
<td>Provide delivery option from existing list of IDs.</td>
</tr>
<tr>
<td>RestrictDeliveryOptions</td>
<td>INVALID_DELIVERY_OPTIONS_STATUS</td>
<td>Provide delivery options in public state.</td>
</tr>
<tr>
<td>RestrictDeliveryOptions</td>
<td>EMPTY_DELIVERY_OPTION_IDS</td>
<td>Provide non-empty list of delivery option IDs.</td>
</tr>
<tr>
<td>RestrictDeliveryOptions</td>
<td>INVALID_MINIMUM_PUBLIC_DELIVERY_OPTIONS</td>
<td>Provide all delivery option IDs.</td>
</tr>
<tr>
<td>RestrictDeliveryOptions</td>
<td>DUPLICATE_DELIVERY_OPTION_IDS</td>
<td>Provide unique delivery option IDs.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INCOMPATIBLE_PRODUCT_STATUS</td>
<td>Use an existing limited or public product.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INCOMPATIBLE_SERVICES</td>
<td>Provide a valid list of compatible services.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>NO_SERVICE_SPECIFIED</td>
<td>Provide at least 1 compatible service.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>DUPLICATE_COMPATIBLE_AWS_SERVICES</td>
<td>Provide unique list of compatible services.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_VERSION_TITLE</td>
<td>Remove spaces before the trademark symbol.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_VERSION_TITLE</td>
<td>Remove the following unsupported characters: [x, y, z]</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_VERSION_TITLE</td>
<td>Remove spaces from the beginning of the version title.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_VERSION_TITLE</td>
<td>Provide version title with fewer than [x] characters.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>DUPLICATE_VERSION_TITLE</td>
<td>The version title must be different from any other version titles of this product.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_RELEASE_NOTES</td>
<td>Remove spaces before the trademark symbol.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_RELEASE_NOTES</td>
<td>Remove unsupported characters: [x, y, z]</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_RELEASE_NOTES</td>
<td>Remove spaces from the beginning of release notes.</td>
</tr>
<tr>
<td>Change type</td>
<td>Error code</td>
<td>Error message</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>-----------------------------------</td>
<td>-------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_RELEASE_NOTES</td>
<td>Provide release notes with fewer than (x) characters.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_USAGE_INSTRUCTIONS</td>
<td>Remove spaces before the trademark symbol.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_USAGE_INSTRUCTIONS</td>
<td>Remove unsupported characters: [x, y, z]</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_USAGE_INSTRUCTIONS</td>
<td>Provide usage instructions with fewer than (x) characters.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_USAGE_INSTRUCTIONS</td>
<td>Provide usage instructions.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>MISSING_CONTAINER_IMAGES</td>
<td>Provide at least 1 container image.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>TOO_MANY_CONTAINER_IMAGES</td>
<td>Provide fewer than 50 container images.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>DUPLICATE_CONTAINER_IMAGES</td>
<td>Provide a unique list of container images.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_CONTAINER_IMAGES</td>
<td>Provide a valid URI for the container image.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_CONTAINER_IMAGE_URI</td>
<td>Provide a valid URI for the container image.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_CONTAINER_IMAGE_TAG</td>
<td>Avoid using 'latest' tag.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>MISSING_DELIVERY_OPTION_IDS</td>
<td>Provide delivery option from existing list of IDs.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>EMPTY_DELIVERY_OPTION_IDS</td>
<td>Provide non-empty list of delivery option IDs.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>DUPLICATE_DELIVERY_OPTION_IDS</td>
<td>Provide unique delivery option IDs.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>DUPLICATE_DELIVERY_OPTION_TITLE</td>
<td>Provide unique delivery option title.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_DELIVERY_OPTION_TITLE</td>
<td>Delivery option title already exists, retry with a different title.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_FULFILLMENT_OPTION_TITLE</td>
<td>Provide delivery option title with fewer than (x) characters.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>DUPLICATE_DELIVERY_OPTION_TITLE</td>
<td>Provide unique delivery option title.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>EMPTY_RESOURCE_NAME</td>
<td>Provide resource name.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>EMPTY_RESOURCE_URL</td>
<td>Provide resource URL.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_RESOURCE_NAME</td>
<td>Provide resource name with fewer than 256 characters.</td>
</tr>
<tr>
<td>Change type</td>
<td>Error code</td>
<td>Error message</td>
</tr>
<tr>
<td>---------------------------</td>
<td>--------------------------</td>
<td>-------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_RESOURCE_URL</td>
<td>Provide resource URL with fewer than 256 characters.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_SHORT_DESCRIPTION</td>
<td>Provide a short description with fewer than 1,000 characters.</td>
</tr>
<tr>
<td>UpdateDeliveryOptions</td>
<td>INVALID_SHORT_DESCRIPTION</td>
<td>Provide short description.</td>
</tr>
</tbody>
</table>
| UpdateDeliveryOptions     | NO_LICENSE_SECRET_KEYS   | For Amazon EKS Anywhere products, provide 1 override parameter for license secret. Needs DefaultValue of 
|                           |                          | "${AWSMP_LICENSE_SECRET}". See example in section.                          |
| UpdateDeliveryOptions     | NO_SERVICE_ACCOUNT_CONFIGURATION | For paid products, provide 1 override parameter for service account configuration. Needs DefaultValue of 
<p>|                           |                          | &quot;${AWSMP_SERVICE_ACCOUNT}&quot;. See example in section.                          |
| UpdateDeliveryOptions     | SCAN_ERROR               | Fix security vulnerability &quot;[y]&quot; on Image &quot;[x]&quot;.                              |
| UpdateDeliveryOptions     | FIELD_NOT_ALLOWED_TO_CHANGE | Field [x] cannot be changed.                                                 |
| UpdateDeliveryOptions     | INVALID_DELIVERY_OPTIONS_STATUS | Provide delivery options in public or limited state.                        |
| UpdateDeliveryOptions     | NO_CHANGE_FOUND          | Provide at least 1 change.                                                   |
| UpdateDeliveryOptions     | MULTIPLE_VERSION_UPDATE  | Provide delivery option IDs from the same version.                           |
| AddRepositories           | INVALID_ECR_REPOSITORY_NAME | Provide repository name in the format: 'nginx-web-app'.                     |
| AddRepositories           | DUPLICATE_ECR_REPOSITORY_NAME | The repository name must be unique.                                         |
| AddRepositories           | MISSING_REPOSITORY_INFORMATION | Provide at least 1 repository name.                                       |
| AddRepositories           | INVALID_ECR_REPOSITORY_NAME | Maximum character length 256 reached. Character length count is inclusive of the seller namespace. |
| UpdateInformation         | MISSING_UPDATES          | Nothing to update. Provide updated information for product.                 |
| UpdateInformation         | MISSING_LOGO_URL         | Provide URL for logo stored in S3.                                           |</p>
<table>
<thead>
<tr>
<th>Change type</th>
<th>Error code</th>
<th>Error message</th>
</tr>
</thead>
<tbody>
<tr>
<td>UpdateInformation</td>
<td>LOGO_COPY_FAILURE</td>
<td>There was an issue copying the logo from S3. Provide a new URL for logo stored in S3.</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_CATEGORY_NAMES</td>
<td>Provide valid category names supported by AWS Marketplace.</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_PRODUCT_TITLE</td>
<td>Remove spaces before trademark symbol.</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_PRODUCT_TITLE</td>
<td>Remove unsupported characters [x, y, z].</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_SHORT_DESCRIPTION</td>
<td>Remove spaces before trademark symbol.</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_SHORT_DESCRIPTION</td>
<td>Remove unsupported characters [x, y, z].</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_LONG_DESCRIPTION</td>
<td>Remove spaces before trademark symbol.</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_LONG_DESCRIPTION</td>
<td>Remove unsupported characters [x, y, z].</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_SUPPORT_DESCRIPTION</td>
<td>Remove spaces before trademark symbol.</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_SUPPORT_DESCRIPTION</td>
<td>Remove unsupported characters [x, y, z].</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_HIGHLIGHTS</td>
<td>Remove spaces before trademark symbol.</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_HIGHLIGHTS</td>
<td>Remove unsupported characters [x, y, z].</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_ADDITIONAL_RESOURCES</td>
<td>Remove spaces before trademark symbol.</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_ADDITIONAL_RESOURCES</td>
<td>Remove unsupported characters [x, y, z].</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_PRODUCT_CATEGORIES</td>
<td>Provide between 1 and 3 product categories.</td>
</tr>
<tr>
<td>UpdateInformation</td>
<td>INVALID_IMAGE_PROPERTIES</td>
<td>Provide an image that meets the product logo requirements. (link)</td>
</tr>
</tbody>
</table>
# Working with a private marketplace

You can use the AWS Marketplace Catalog API to manage a *private marketplace* for your AWS account or organization.

For more information about private marketplaces, see [Private marketplaces](#) in the *AWS Marketplace Buyer Guide*.

The following table details a set of tasks to manage private marketplaces and the change types that apply to each task.

<table>
<thead>
<tr>
<th>Task</th>
<th>Action</th>
<th>Change types</th>
</tr>
</thead>
<tbody>
<tr>
<td>the section called “Creating a private marketplace” (p. 56)</td>
<td>StartChangeSet</td>
<td>CreateExperience</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CreateProcurementPolicy</td>
</tr>
<tr>
<td>the section called “Changing the branding of a private marketplace experience” (p. 59)</td>
<td>StartChangeSet</td>
<td>CreateBrandingSettings</td>
</tr>
<tr>
<td></td>
<td></td>
<td>UpdateBrandingSettings</td>
</tr>
<tr>
<td>the section called “Enabling or disabling a private marketplace experience” (p. 61)</td>
<td>StartChangeSet</td>
<td>UpdateProcurementPolicy</td>
</tr>
<tr>
<td>the section called “Getting a list of products in a private marketplace experience” (p. 62)</td>
<td>DescribeEntity</td>
<td>Not applicable</td>
</tr>
<tr>
<td>the section called “Adding or removing products from a private marketplace” (p. 64)</td>
<td>StartChangeSet</td>
<td>AllowProductProcurement</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DenyProductProcurement</td>
</tr>
<tr>
<td>the section called “Finding products” (p. 65)</td>
<td>Not applicable</td>
<td>Not applicable</td>
</tr>
<tr>
<td>the section called “Working with private marketplaces for AWS Organizations” (p. 65)</td>
<td>Not applicable</td>
<td>Not applicable</td>
</tr>
<tr>
<td>the section called “Associating accounts to experiences” (p. 66)</td>
<td>StartChangeSet</td>
<td>AssociateAudience</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DisassociateAudience</td>
</tr>
<tr>
<td>Archiving and reactivating a private marketplace experience (p. 68)</td>
<td>StartChangeSet</td>
<td>RestrictExperience</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ReviveExperience</td>
</tr>
</tbody>
</table>

## Creating a private marketplace

A private marketplace for an AWS account can be thought of as a list of products that users are allowed to procure in that account, and branding for the marketplace. In an organization with multiple accounts,
the accounts can be grouped together and share experiences. For example, you could have one set of products that all accounts in the organization are allowed to procure, or you could have a different list of products for each account in the organization. Each list of approved products and branding is called a procurement experience.

In the AWS Marketplace Catalog API, four entities represent an experience:

- **Experience entity** – This entity is at the highest level of the experience and contains two child entities.
- **ProcurementPolicy entity** – This entity represents the products that have been allowed and denied in your private marketplace.
- **BrandingSettings entity** – You can also create a BrandingSettings entity to define how your private marketplace looks to your users.
- **Audience entity** – You must also associate one or more Audience entities, which define the set of AWS accounts that the experience applies to.

The steps to create a procurement experience are as follows:

1. Create the Experience entity.
2. Create a ProcurementPolicy entity to store the list of products that are allowed or denied for the experience.
3. (Optional) Create a BrandingSettings entity to customize the look of your marketplace experience.
4. Associate accounts with your experience. For single AWS accounts, this is just the one account.
5. Enable the experience.

**Note**
If your account is part of an organization in AWS Organizations, see [Working with private marketplaces for AWS Organizations](p. 65).

**Create the Experience entity**

To create the Experience entity, use the StartChangeSet action with the CreateExperience value for the ChangeType parameter to request that the experience be created by AWS Marketplace. See the following code example.

```json
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
   "Catalog": "AWSMarketplace",
   "ChangeSet": [
      {
         "ChangeType": "CreateExperience",
         "DetailsDocument": {
            "Name": "ExamplePrivateMarketplace"
         },
         "Entity": {
            "Type": "Experience@1.0"
         }
      }
   ],
   "ChangeSetName": "Create Private Marketplace Example"
}
```

57
In this action, Entity is a template for the entity that you want to create. It is assigned an EntityId when it is created. ChangeSetName identifies the change to help you find it later.

The response looks like the following.

```json
{
  "ChangeSetArn": "arn:...:AWSMarketplace/ChangeSet/abcd1234example5678frjzkz",
  "ChangeSetId": "abcd1234example5678frjzkz"
}
```

The response includes a ChangeSetId that you can use to get the status of your change request as it is processed with DescribeChangeSet. You can also use ListEntities to find your Experience entity without the ChangeSetId. For more information about change sets, see [Working with change sets](p. 3).

A newly created Experience entity doesn't have a procurement policy by default. It is also created with default settings for branding. For more information about branding settings, including how to customize them, see [Changing the branding of a private marketplace experience](p. 59).

**Create a ProcurementPolicy entity**

You must create a ProcurementPolicy entity. By default, a new Experience entity is disabled, so you can create the procurement policy before enabling it.

**Note**

An Experience entity with no procurement policy (null) allows all products to be procured in your private marketplace. An Experience entity with an empty procurement policy has no products available to users to procure.

To allow and deny products in your private marketplace, you must create the procurement policy. To do this, you again call StartChangeSet, but this time with the ChangeType of CreateProcurementPolicy. The following code example creates an empty procurement policy.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "CreateProcurementPolicy",
      "DetailsDocument": {
        "Name": "ExampleProcurementPolicy"
      },
      "Entity": {
        "Type": "Experience@1.0",
        "Identifier": "exp-1234example@1"
      }
    }
  ]
}
```

The Entity you provide in this action is the Experience entity that you want the procurement policy created within, so you must include the identifier for the entity that you created earlier. Use ListEntities to find the Experience entity. You can also return the identifier by using DescribeChangeSet with the change set identifier from the CreateExperience action.
Changing the branding of a private marketplace experience

You can customize the look of your private marketplace for your users. Without customization, your private marketplace will have the default branding settings, which are described below. Aspects of branding that you can change in a private marketplace include the following:

- **Title** – The name displayed for your private marketplace. This is the same as the Name field in the private marketplace Profile settings screen. If you set the Title to Example, then the text displayed is Example Private Marketplace.

- **Information** – The paragraph displayed under the name in your private marketplace. This is the same as the Description field in Profile settings. The default is no information, in which case a general description of private marketplaces is displayed.

- **ThemeColor** – The color displayed in the banner of your private marketplace. This is a color in RGB hexadecimal format. This value is the same as the Theme color field in Profile settings. The default value is #232F3E.

- **LogoUrl** – The URL that points to an image file to be used as the logo on your private marketplace. The URL must be publicly available (for example, a signed Amazon S3 URL). The file must be either a .png or .svg file and be under 500kb. If necessary, the image file will be resized to a maximum height of 30 pixels and a maximum width of 100 pixels. This is the same value as the Logo Select in Profile Settings. The default is to not show a logo.

To set these values, you must first create a BrandingSettings entity with the CreateBrandingSettings change type. You can then request an UpdateBrandingSettings change to set or change the branding. You only need to create a BrandingSettings object once. To create this object, call StartChangeSet with the CreateBrandingSettings change type, as shown in the following code example.

```plaintext
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
    "Catalog": "AWSMarketplace",
}
```
Changing the branding of a private marketplace experience

```
"ChangeSet": [ 
  { 
    "ChangeType": "CreateBrandingSettings",
    "DetailsDocument": { 
      "Name": "ExampleBrandingSettingsName"
    },
    "Entity": { 
      "Type": "Experience@1.0",
      "Identifier": "exp-1234example@2"
    }
  }
]
```

This example modifies the Experience entity by adding the BrandingSettings object to it. The revision of the entity identifier has incremented to 2. For more information about revisions for identifiers, see Identifier (p. 2).

**Note**
You can specify all the details of the branding settings in the call to create the branding settings entity. The details facet is the same for CreateBrandingSettings and UpdateBrandingSettings.

You modify the settings by calling StartChangeSet with the UpdateBrandingSettings change type. The settings are part of the Configuration of the DetailsDocument object.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [ 
    { 
      "ChangeType": "UpdateBrandingSettings",
      "DetailsDocument": { 
        "Name": "ExampleBrandingSettingsName",
        "Description": "Example description",
        "Configuration": { 
          "Title": "ExampleName",
          "Information": "Example description.",
          "ThemeColor": "#0e7f74",
          "LogoUrl": "https://example.com/path/mylogo.png"
        }
      },
      "Entity": { 
        "Type": "Experience@1.0",
        "Identifier": "exp-1234example@3"
      }
    }
  ]
}
```

**Note**
The URL for the logo is used to make a copy during the update change. After the change is complete, if you remove or change the URL at that path, it will not affect your private marketplace unless you again request UpdateBrandingSettings.
Enabling or disabling a private marketplace experience

When a private marketplace is enabled (and has a procurement policy), users in associated accounts can only purchase products that you have approved. When no private marketplace experience is enabled for an account, users can purchase products across the full AWS Marketplace catalog.

To enable a private marketplace, use the `StartChangeRequest` action with the `UpdateExperience` change type.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateExperience",
      "DetailsDocument": {
        "Status": "Enabled"
      },
      "Entity": {
        "Type": "Experience@1.0",
        "Identifier": "exp-1234example@4"
      }
    }
  ]
}
```

Similarly, you can use the same action and `ChangeType`, but change the `Status` in `DetailsDocument` to `Disabled` to disable a private marketplace.

**Note**
Disabling a private marketplace keeps your list of both allowed and denied products, as well as customizations, such as branding. When a private marketplace is disabled, users no longer see the private marketplace (although they may still be governed by the management account experience). If there are no private marketplace experiences enabled for an account, then all restrictions are removed, and users are able to procure any products in the public AWS Marketplace.

Enabling or disabling user requests

Users in your organization can view the full public AWS Marketplace, but they can only subscribe to the products that you have allowed. By default, they can request that a product that is not in the private marketplace be added to it. These requests show up in the private marketplace administrator page (`Private Marketplace`), where you can decide whether to accept or deny the request (and whether to block further requests for the same product). You cannot see or respond to the requests by using the Catalog API.

You can enable or disable the ability for users to create requests for your private marketplace experience. Use `StartChangeSet` with the `UpdateProcurementPolicy` change type. The ability to make requests is disabled in the following code example.
POST /StartChangeSet HTTP/1.1
Content-type: application/json
{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "ChangeType": "UpdateProcurementPolicy",
      "DetailsDocument": {
        "Configuration": {
          "PolicyResourceRequests": "Deny"
        },
        "Entity": {
          "Type": "Experience@1.0",
          "Identifier": "exp-1234example@5"
        }
      }
    }
  ]
}

To enable the change request capability for users, use Allow instead of Deny in PolicyResourceRequests.

To learn how to get the current status of this setting, see the next section, Getting a list of products in a private marketplace experience (p. 62).

Getting a list of products in a private marketplace experience

The products allowed (and denied) in a private marketplace are part of the procurement policy in the Experience entity. To get the details about the procurement policies in a private marketplace, you first get the procurement policy identifier from the Experience entity, and then call DescribeEntity with that identifier.

To get the procurement policy identifier, use DescribeEntity on the Experience entity that you are interested in, as shown in the following command.

GET /DescribeEntity?catalog=AWSMarketplace&entityId=exp-example01

Following is an example response.

{
  "Details": "\"Name\": \"New Private Marketplace\", \"Status\": \"Enabled\", \"ProcurementPolicies\": \"procpolicy-123example456\", \"BrandingSettings\": \"brandsettings-456example123\"",
  "DetailsDocument": {
    "Name": "New Private Marketplace",
    "Status": "Enabled",
    "ProcurementPolicies": [
      "procpolicy-123example456"
    ]
  }
}
Note
The DetailsDocument attribute contains the entity details as a JSON object. The legacy Details attribute contains the same JSON object as a string.

You can use the returned EntityId for the procurement policy to get the details, as shown in the following command.

GET /DescribeEntity?catalog=AWSMarketplace&entityId=procpolicy-123example456

This returns the full details of the policy, including both allowed and denied products. Following is an example response.

```json
{
  "Details": ""{
    "Name": "ExampleProcurementPolicy",
    "Statements": [
      {
        "Effect": "Allow",
        "Resources": [
          {
            "Type": "Product",
            "Ids": [
              "example1-1234-abcd-5678-90abcdef1234"
            ]
          },
          {
            "Type": "Product",
            "Ids": [
              "example2-2345-bcde-6789-01bcdea2345"
            ]
          }
        ]
      },
      {
        "Effect": "Deny",
        "Resources": [
          {
            "Type": "Product",
            "Ids": [
              "example3-3456-cdef-7890-12defabc5678"
            ]
          }
        ]
      }
    ]
  },
  "DetailsDocument": {
    "Name": "ExampleProcurementPolicy",
    "Statements": [
      {
        "Effect": "Allow",
        "Resources": [
          {
            "Type": "Product",
            "Ids": [
              "example1-1234-abcd-5678-90abcdef1234"
            ]
          },
          {
            "Type": "Product",
            "Ids": [
              "example2-2345-bcde-6789-01bcdea2345"
            ]
          }
        ]
      },
      {
        "Effect": "Deny",
        "Resources": [
          {
            "Type": "Product",
            "Ids": [
              "example3-3456-cdef-7890-12defabc5678"
            ]
          }
        ]
      }
    ]
  }
}
```
Adding or removing products from a private marketplace

By default, a private marketplace does not have any approved products in it. Use change requests to add or remove a product. To add a product, use the AllowProductProcurement change type. To remove a product, use the DenyProductProcurement change type.

The following code example shows the AllowProductProcurement change type with the StartChangeSet action to add a product to a private marketplace.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
   "Catalog": "AWSMarketplace",
   "ChangeSet": [
      {
         "ChangeType": "AllowProductProcurement",
         "DetailsDocument": {
            "Products": [
               {
                  "Ids": [
                     "example-1234-abcd-5678-90abcded1234"
                  ],
                  "Notes": "Useful product"
               }
            ],
            "Entity": {
               "Identifier": "exp-1234example@6",
               "Type": "Experience@1.0"
            }
         }
      }
   ]
}
```

In this example, the procurement policy has two allowed products and one denied product. The policy allows user resource requests.
You add the product to the Experience entity for a private marketplace by using AllowProductProcurement. The syntax to remove a product from a private marketplace is identical, with the exception that you use the DenyProductProcurement ChangeType instead of AllowProductProcurement. The products are added to the allow (or deny) list of the ProcurementPolicy entity that is contained by your Experience entity.

**Note**

The list of products in the DetailsDocument of your change is an array of Ids, so you can add (or remove) multiple products with one call by including a list of product identifiers. The limit is 50 products in a single request.

The Notes field for the list of Ids is not required. However, you can use it to record why a decision to allow or deny a set of products was made.

### Finding products

By getting the details of your procurement policy, you can find the product IDs for the products that are already in a private marketplace. However, the AWS Marketplace Catalog API does not provide a way to find the product IDs for other products. There are two ways to get product IDs to use with the Catalog API service:

- **Public marketplace** – After you find a product in the public marketplace, choose **Continue to Subscribe** to see a details page about the product (it will not subscribe you to the product). The URL will include the product ID as a parameter. For example, in the URL `https://aws.amazon.com/marketplace/fulfillment?productId=ab1234cd-1234-abcd-5678-90abcdef1234&ref_=dtl_psb_continue`, `ab1234cd-1234-abcd-5678-90abcdef1234` is the product ID.

- **AWS Marketplace Discovery API** – Programmatically, you can access the full list of products in the AWS Marketplace by using the Discovery API. The Discovery API is a private API. You must request access to be able to use it. For more information, see [Getting access to the Discovery API (p. 80)](#).

### Working with private marketplaces for AWS Organizations

Whether you are working with a private marketplace for your account or your organization, you use the same API. However, there are differences when working within your organization:

- You must create the first Experience of your private marketplace from within the management account. This creates the Experience entity and enables the private marketplace feature for your organization.

- After the first Experience is created, it is automatically shared, allowing you to edit and update it from any account in the organization, as long as you have the correct permissions. You can also create additional Experience entities from any account.

- When listing objects in a private marketplace from another account (that has been automatically shared across the organization), you must specifically request them with the SharedWithMe filter. This applies to both ListEntities and ListChangeSets actions.

For example, when you call ListEntities from an account that isn't the management account, it only returns entities created by or owned by that account, by default. To list Experience objects in your own account, call ListEntities as shown in the following code example.
POST /ListEntities HTTP/1.1
Content-Type: application/json
{
  "Catalog":"AWSMarketplace",
  "EntityType":"Experience"
}

However, to list the entities that have been shared with you, you must add a FilterList with a Scope of SharedWithMe, as shown in the following code example. As a result, AWS Marketplace searches outside of your own account to find entities that are shared with you.

POST /ListEntities HTTP/1.1
Content-Type: application/json
{
  "Catalog":"AWSMarketplace",
  "EntityType":"Experience",
  "FilterList":
  [
    {
      "Name": "Scope",
      "ValueList":
      [
        "SharedWithMe"
      ]
    }
  ]
}

In this case, only entities outside of your account (the ones for your organization) are returned.

Similarly, to call ListChangeSets, you must set the scope, as shown in the following code example.

POST /ListChangeSets HTTP/1.1
Content-Type: application/json
{
  "Catalog":"AWSMarketplace",
  "FilterList":
  [
    {
      "Name": "Scope",
      "ValueList":
      [
        "SharedWithMe"
      ]
    }
  ]
}

This returns change sets that apply to a shared private marketplace for your organization.

## Associating accounts to experiences

A private marketplace experience must have one or more accounts associated with it in order to have any effects in your organization. For a single AWS account, you must associate the account with the experience to use the private marketplace. In an organization, you can have multiple experiences apply to different accounts.

**Note**

The experience that is associated with the management account is the default for all other accounts in the organization. Associating a linked account with a different experience directly sets a different experience for that account.

To associate an account to an experience, use the AssociateAudience change type with the StartChangeSet action, as shown in the following code example.

POST /StartChangeSet HTTP/1.1
Content-type: application/json
The audience is the list of principals that are associated with the Experience. A principal is an AWS account, defined by its ID. Principals is a list, so you can include multiple accounts to be associated with the experience. After the first call, subsequent calls to the AssociateAudience change type will add principals to the association for the experience.

You can also remove accounts from an experience. Use the DisassociateAudience change type to do this, as shown in the following code example. Use the DisassociateAudience change type to do this, as shown in the following code example.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet": [
    {
      "Entity": {
        "Type": "Experience@1.0",
        "Identifier": "exp-example01@02"
      },
      "ChangeType": "DisassociateAudience",
      "DetailsDocument": {
        "Principals": [
          "012345678901"
        ]
      }
    }
  ],
  "ChangeSetName": "Disassociate audience example"
}
```

**Note**

An account can only be directly associated with one experience. To move an account from being directly associated with one experience to another experience, you must disassociate it from the initial experience, then associate it with the second.
Archiving and reactivating a private marketplace experience

You can remove a private marketplace experience by archiving it. Archived experiences can’t be updated or used to govern accounts in your organization. If you have account groups associated with an archived experience, you can associate them with a different experience. If you decide to use the experience at a later time, you can always reactivate it. Administrators from the account that created the experience have permissions to archive and reactivate experiences.

**Note**
Before archiving an experience, you must disable it. For information about disabling an experience, see [Configuring your private marketplace](#) in the *AWS Marketplace Buyer Guide*.

To archive an experience, use the `RestrictExperience` change type with the `StartChangeSet` action, as shown in the following code example.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "RestrictExperience",
      "DetailsDocument": {},
      "Entity":
      {
        "Type": "Experience@1.0",
        "Identifier": "exp-1234example"
      }
    }
  ]
}
```

To reactivate an experience, use the `ReviveExperience` change type with the `StartChangeSet` action, as shown in the following code example.

```
POST /StartChangeSet HTTP/1.1
Content-type: application/json

{
  "Catalog": "AWSMarketplace",
  "ChangeSet":
  [
    {
      "ChangeType": "ReviveExperience",
      "DetailsDocument": {},
      "Entity":
      {
        "Type": "Experience@1.0",
        "Identifier": "exp-1234example"
      }
    }
  ]
}
```
## Errors in the private marketplace API

The following errors are specific to the private marketplace actions in the AWS Marketplace Catalog API.

<table>
<thead>
<tr>
<th>Change type</th>
<th>Error code</th>
<th>Error message</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Errors returned directly by the StartChangeSet action</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>All</td>
<td>422</td>
<td>Document not valid JSON format</td>
<td>Invalid JSON input used, check your syntax.</td>
</tr>
<tr>
<td>AllowProductProcurement, DenyProductProcurement</td>
<td>422</td>
<td>Values in Ids array must be unique</td>
<td>You can't include the same product multiple times in a single change request.</td>
</tr>
<tr>
<td>AllowProductProcurement, DenyProductProcurement</td>
<td>422</td>
<td>Cumulative number of values in Ids array must be less than or equal to 50</td>
<td>You can allow or deny up to 50 products in a single change request.</td>
</tr>
<tr>
<td><strong>Errors found by calling the DescribeChangeSet action</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CreateBrandingSettings, UpdateBrandingSettings</td>
<td>INVALID_URL</td>
<td>Image could not be fetched from the input URL</td>
<td>You must specify a valid, reachable URL for the logo field in BrandingSettings.</td>
</tr>
<tr>
<td>CreateBrandingSettings, UpdateBrandingSettings</td>
<td>INVALID_IMAGE</td>
<td>Image verification for type, content, or file size failed. Only .png and .svg file types with sizes less than or equal to 500KB are supported.</td>
<td>Your image file must match the logo requirements for branding settings.</td>
</tr>
<tr>
<td>AllowProductProcurement, DenyProductProcurement</td>
<td>ENTITY_NOT_FOUND</td>
<td>Procurement policy missing from Experience</td>
<td>You must create a ProcurementPolicy before allowing or denying products.</td>
</tr>
<tr>
<td>CreateProcurementPolicy</td>
<td>ENTITY_ALREADY_EXISTS</td>
<td>Procurement policy exists for Experience</td>
<td>You can only have a single procurement policy for a private marketplace.</td>
</tr>
<tr>
<td>UpdateProcurementPolicy</td>
<td>ENTITY_NOT_FOUND</td>
<td>Procurement policy missing from Experience</td>
<td>You must create a ProcurementPolicy before updating the procurement policy.</td>
</tr>
<tr>
<td>CreateBrandingSettings</td>
<td>ENTITY_ALREADY_EXISTS</td>
<td>Branding settings exists for Experience</td>
<td>You can only have a single branding settings for a private marketplace.</td>
</tr>
<tr>
<td>UpdateBrandingSettings</td>
<td>ENTITY_NOT_FOUND</td>
<td>Branding settings missing from Experience</td>
<td>You must create a BrandingSettings</td>
</tr>
<tr>
<td>Change type</td>
<td>Error code</td>
<td>Error message</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>---------------------------</td>
<td>-------------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>AssociateAudience</td>
<td>CALLER_NOT_AUTHORIZED</td>
<td>Caller not authorized to execute the action</td>
<td>You must have permissions to call the action. The accounts being added must be in the same organization.</td>
</tr>
<tr>
<td>CreateExperience</td>
<td>CALLER_NOT_AUTHORIZED</td>
<td>Caller not authorized to create experience. Initial experience must be created from management account, if AWS Organizations is enabled.</td>
<td>You must have permissions to create an experience, and the initial experience for an organization must be created in the management account.</td>
</tr>
<tr>
<td>AssociateAudience</td>
<td>ENTITY_ALREADY_EXISTS</td>
<td>An experience is already associated with the account {accountId}. Disassociate previous experience before updating</td>
<td>You can only associate a single experience with an account. Disassociate the current experience before associating a new one.</td>
</tr>
<tr>
<td>AssociateAudience, DisassociateAudience</td>
<td>ENTITY_IN_USE</td>
<td>There is already a conflicting change in progress for the selected account. Try again later</td>
<td>You can't change the association with an account while another change request to change the association is already in progress.</td>
</tr>
</tbody>
</table>

### Entity types defined by private marketplace

The following table lists the private marketplace entity types, purpose, and actions on which each can be specified. Each entity type can be used to specify a resource Amazon Resource Name (ARN) that can be used in the AWS Identity and Access Management (IAM) policy. For more details on ARN formats, see the **Catalog API entities** (p. 1).

<table>
<thead>
<tr>
<th>Entity</th>
<th>Purpose</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Experience</td>
<td>Stores the top-level settings for a private marketplace</td>
<td>StartChangeSet</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DescribeEntity</td>
</tr>
<tr>
<td>BrandingSettings</td>
<td>Stores the branding settings for a private marketplace</td>
<td>DescribeEntity</td>
</tr>
<tr>
<td>ProcurementPolicy</td>
<td>Stores the procurement settings and lists of products in a private marketplace</td>
<td>DescribeEntity</td>
</tr>
<tr>
<td>Audience</td>
<td>Stores the details of accounts associated with a private marketplace</td>
<td>DescribeEntity</td>
</tr>
</tbody>
</table>
Working with AWS RAM to share resources

AWS Marketplace Catalog API integrates with AWS Resource Access Manager (AWS RAM) to enable resource sharing. A resource is an entity that users can work with in AWS Marketplace, such as a product, an offer, or a resale authorization. With AWS RAM, you can share some AWS Marketplace Catalog API resources with other AWS accounts. You share resources that you own by creating a resource share. A resource share specifies the resources that you want to share and the consumers with whom to share them.

Contents
- Prerequisites for sharing AWS Marketplace entities (p. 71)
- Sharing an AWS Marketplace entity (p. 71)

Prerequisites for sharing AWS Marketplace entities

Before sharing entities in AWS Marketplace Catalog API, you must meet the following prerequisites:

- You can only have one resource policy attached to your AWS Marketplace entity.
- To share an AWS Marketplace entity, you must own it in your AWS account. This requirement means that the entity must be allocated or provisioned in your account. You can't share an AWS Marketplace entity that has been shared with you.

Sharing an AWS Marketplace entity

With AWS Marketplace resource sharing, entity owners can share their entities with other AWS accounts in AWS Marketplace. Entity-owners can be ISVs and channel partners. Entities that can be shared are products, offers, and resale authorizations.

Note
At this time, you can only share entities. Entities in AWS Marketplace include AmiProduct, Audience, BrandingSettings, ContainerProduct, Experience, and ProcurementPolicy.

For more information about AWS RAM, see the AWS RAM User Guide. For more information about managing your shared resources, see Using shared AWS resources in the AWS RAM User Guide.

As a sharing account, you can set read-only or both read/write on the resources that you want to share. These permissions determine what operations a consuming account can perform on the resources that are shared with them.

- Sharing account – The resource that is shared and in which the AWS RAM administrator creates the AWS resource share by using AWS RAM.
- Consuming account – The AWS account to which a resource is shared. The resource share can specify an entire account as the principal, or for some resource types, individual roles or users in the account.

To share an AWS Marketplace entity, you must add it to a resource share. A resource share is an AWS RAM resource that lets you share your resources across AWS accounts. A resource share specifies the
resources to share, and the consumers with whom they are shared. When you share an entity using the
AWS Marketplace console, you add it to an existing resource share. To add the AWS Marketplace entity to
a new resource share, you must first create the resource share using the AWS RAM console.

You can share an AWS Marketplace entity that you own using the AWS Marketplace console, AWS RAM
console, or the AWS Command Line Interface (AWS CLI).

**To share an AWS Marketplace entity that you own using the AWS RAM console**


**To share an AWS Marketplace entity that you own using the AWS CLI**

Use the `create-resource-share` command.

**Note**

For resource types such as entities that support resource-based policies, you can use AWS RAM
to share resources to use additional AWS RAM features. For more information, see [Resource-based policy](https://docs.aws.amazon.com/rms/latest/userguide/resource-share-resource-based-policies.html) in the *AWS RAM User Guide*. AWS RAM uses the AWS Marketplace Catalog API to automatically construct the resource policy from permissions in a resource share and manages that resource policy for you.

For information about how to set, view, or delete AWS resource-based policies on your AWS Marketplace
to share resources through AWS RAM, see [Allowing actions on all resources](https://docs.aws.amazon.com/rms/latest/userguide/resource-share-resource-based-policies.html) in the *AWS RAM User Guide*.

**Differences between sharing an entity through AWS RAM and the AWS Marketplace Catalog API**

In addition to sharing your entity through AWS RAM, you can also set, view, or delete AWS resource-
based policies on your entities through the AWS Marketplace Catalog API. However, there are a few
differences between sharing your entity through AWS RAM and through the AWS Marketplace Catalog
API.

When you share an entity through AWS RAM:

- If you share your entity with accounts that are outside of AWS Organizations, the consuming account
must first accept your sharing request before the entity is shared.
- The consuming account can discover the shared entity through `ListEntities` with `OwnershipType`
set to `SHARED`.
- You must adhere to several resource quotas. For more information, see [Service quotas for AWS RAM](https://docs.aws.amazon.com/rms/latest/userguide/service-quotas.html) in the *AWS RAM User Guide*.

When you share an entity through the AWS Marketplace Catalog API:

- Your entity will be shared as soon as the `PutResourcePolicy` request succeeds with no input from
the consuming account.
- The consuming account can't discover the shared entity through `ListEntities` with
`OwnershipType` set to `SHARED`. Instead, the owner of the sharing account must inform the
consuming account of the shared entity ID.

**Note**

If your use case requires sharing resources that might exceed AWS RAM service quotas, or if you
want to share resources without direct input from the consuming account, consider sharing
through the AWS Marketplace Catalog API. For all other use cases, consider using AWS RAM to
share AWS Marketplace resources.
The following sections detail how you can set, view, or delete AWS resource-based policies on your
entities through the AWS Marketplace Catalog API.

Topics

• Attach read-only policy to your resource (p. 73)
• Attach read and write resource policy to your resource (p. 73)
• View resource policy set on your resource (p. 74)
• Delete resource policy on your resource (p. 75)
• View all resources owned by you and shared with you (p. 75)

Attach read-only policy to your resource

You can create a read-only resource-based policy on your shared resource using a sharing account. With
this policy, the principal can only view the details of the resource that is shared with them.

Request

```plaintext
POST /PutResourcePolicy HTTP/1.1
Content-type: application/json

{
  "Policy": {
    "Version": "2012-10-17",
    "Statement": {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::222233334444:root"
      }
    },
    "Action": ["aws-marketplace:DescribeEntity"],
    "Resource": ["arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/AmiProduct/example2-abcd-1234-5ef6"
  ]
}
```

Response

```plaintext
HTTP/1.1 200
Content-type: application/json

{}
```

Attach read and write resource policy to your resource

As a sharing account, you can create a read and write resource-based policy on your shared resource.
With this policy, the principal can view the details and perform write operations on the resource that is
shared with them.

Request

```plaintext
POST /PutResourcePolicy HTTP/1.1
Content-type: application/json

{
  "Policy": {
    "Version": "2012-10-17",
    "Statement": {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::222233334444:root"
      }
    },
    "Resource": ["arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/AmiProduct/example2-abcd-1234-5ef6"
  ]
}
```
View resource policy set on your resource

As a sharing account, you can view the resource policy that is set on your shared resource.

Request

POST /GetResourcePolicy HTTP/1.1
Content-type: application/json


Response

HTTP/1.1 200
Content-type: application/json

{"Policy": {
"Version": "2012-10-17",
"Statement": {
"Effect": "Allow",
"Principal": {
"AWS": "arn:aws:iam::222233334444:root"
},
"Action": [
"aws-marketplace:DescribeEntity",
"aws-marketplace:StartChangeSet"
],
"Resource": [
"arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/AmiProduct/example2-abcd-1234-5ef6"
]
}}
Delete resource policy on your resource

As a sharing account, you can delete the resource policy that is set on your shared resource.

Request

POST /DeleteResourcePolicy HTTP/1.1
Content-type: application/json

{
    "ResourceArn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/AmiProduct/example2-abcd-1234-5ef6"
}

Response

HTTP/1.1 200
Content-type: application/json

{}

View all resources owned by you and shared with you

As a consuming account, you can view the resources that are shared with you.

Note
You can view the resources shared with you only if the resources were shared through AWS RAM.

Request

POST /ListEntities HTTP/1.1
Content-type: application/json

{
    "Catalog": "AWSMarketplace",
    "EntityType": "AmiProduct",
    "FilterList": [
        {
            "Name": "EntityId",
            "ValueList": [ "example2-abcd-1234-5ef6" ]
        }
    ],
    "OwnershipType": "SHARED"
}

Response
HTTP/1.1 200
Content-type: application/json

{
    "EntitySummaryList": [
        {
            "EntityId": "example2-abcd-1234-5ef6",
            "EntityType": "AmiProduct",
            "LastModifiedDate": "2018-02-27T13:45:22Z",
            "Name": "TestProduct",
            "Visibility": "public"
        }
    ],
    "NextToken": ""
}
Logging and notifications

The AWS Marketplace Catalog API supports logging with AWS CloudTrail and notifications with Amazon EventBridge.

Topics
- Logging AWS Marketplace Catalog API calls with CloudTrail (p. 77)
- AWS Marketplace Catalog API Amazon EventBridge events (p. 79)

Logging AWS Marketplace Catalog API calls with CloudTrail

The AWS Marketplace Catalog API is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service. CloudTrail captures all calls to the Catalog API as events, including calls from the AWS Marketplace Management Portal.

If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon Simple Storage Service (Amazon S3) bucket. 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, the IP address from which the request was made, who made the request, when it was made, and additional details.

AWS Marketplace Catalog API information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When activity occurs in the AWS Marketplace Catalog API, that 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, create a trail. A trail enables CloudTrail to deliver log files to an Amazon S3 bucket. By default, when you create a trail in the console, the trail applies to all AWS Regions. The trail logs events from all AWS Regions in the AWS partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, 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 Marketplace Catalog API actions are logged by CloudTrail and are documented in this API Reference. For example, calls to the StartChangeSet, DescribeChangeSet, and ListChangeSets functions are logged.
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 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 in the AWS CloudTrail User Guide.

Understanding AWS Marketplace catalog log file entries

A trail is a configuration that enables delivery of events as log files to an Amazon 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 specific order.

**Note**

These examples have been formatted for improved readability. In a CloudTrail log file, all entries and events are concatenated into a single line. In addition, this example has been limited to a single AWS Marketplace Catalog API entry. In a real CloudTrail log file, you see entries and events from multiple AWS services.

The following example shows a AWS Marketplace Catalog API log entry that demonstrates the ListEntities action:

```json
[
  {
    "eventVersion": "1.05",
    "userIdentity": {
      "type": "IAMUser",
      "principalId": "ABCDEFGHIJKLMNOP12345",
      "arn": "arn:aws:iam::123456789010:principal/CloudTrailTestUser",
      "accountId": "123456789010",
      "accessKeyId": "ABCDEFGHIJKLMNOP1234",
      "userName": "CloudTrailTestUser"
    },
    "eventTime": "2019-10-17T21:49:23Z",
    "eventSource": "marketplacecatalog.amazonaws.com",
    "eventName": "ListEntities",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "127.0.0.1",
    "userAgent": "PostmanRuntime/7.18.0",
    "requestParameters": {
      "catalog": "AWSMarketplace",
      "entityType": "EntityProduct",
      "sort": {
        "sortBy": "LastUpdateTimeInMillis",
        "sortOrder": "DESC"
      },
      "maxResults": 20
    },
    "responseElements": null,
    "requestID": "fEXAMPLE-cb3e-4e21-86fd-6b3EXAMPLEd1",
    "eventType": "AwsApiCall",
    "recipientAccountId": "123456789010"
  }
]
```
AWS Marketplace Catalog API Amazon EventBridge events

AWS Marketplace is integrated with Amazon EventBridge, formerly called Amazon CloudWatch Events. EventBridge is an event bus service that you can use to connect your applications with data from a variety of sources.

For information on how sellers, channel partners, and private marketplace administrators can receive ChangeSet status events using EventBridge, see Events for change sets in the AWS Marketplace Seller Guide.
AWS Marketplace Discovery API

The AWS Marketplace Catalog API service provides an API interface to manage the products that you create as a seller in AWS Marketplace, and to manage your private marketplace. However, for discovery use cases, you must use the AWS Marketplace Discovery API (Discovery API).

The Discovery API enables your buyers to have a frictionless discovery experience of AWS Marketplace listings on your web properties. You can use the Discovery API to:

- Get listing information from AWS Marketplace, such as long and short product descriptions, marketplace categories, badges, media, pricing model/unit, information, publisher, reviews, and more.
- Populate your website, platform, and private marketplace with AWS Marketplace listings.
- Create a custom view of AWS Marketplace listings for your customers where you show offerings from other sellers and offer value-added functionality.

With the Discovery API, you can create a browse and search functionality of the entire AWS Marketplace catalog or a curated view that best serves your customers. This dynamic experience helps direct your customers to the most up-to-date and relevant AWS Marketplace pages to purchase the products they need. Additionally, you can access the full list of AWS Marketplace products to help populate your private marketplace.

**Note**
The Discovery API is currently available to select AWS customers and is available upon request to, and approval by, the AWS Marketplace Team. For more information, see the next section, Getting access to the Discovery API (p. 80).

Getting access to the Discovery API

To request access, or to get answers to questions about the Discovery API, reach out to your current AWS Marketplace contact. If you don't have an AWS Marketplace contact, you don't know who your contact is, or you don't have an AWS Marketplace Seller account, submit a general inquiry to the AWS Marketplace Seller Operations team.

After you receive access to the Discovery API, you will receive a unique Integration ID, API documentation, and a SDK to help you integrate with and call the Discovery API.

Logging Discovery API calls using AWS CloudTrail

The Discovery API is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in AWS Marketplace. CloudTrail captures all API calls for the Discovery API as events. The calls captured include calls from the AWS Marketplace console and code calls to the Discovery API operations.

If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for the Discovery API. A trail enables CloudTrail to deliver log files to an Amazon S3 bucket. 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 the Discovery API, the IP address from which the request was made, who made the request, when it was made, and additional details.

For more information about CloudTrail, see the AWS CloudTrail User Guide.
Discovery API information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When activity occurs in the Discovery API, that 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 the Discovery API, create a trail. A trail enables CloudTrail to deliver log files to an Amazon S3 bucket. By default, when you create a trail in the console, the trail applies to all AWS Regions. The trail logs events from all Regions in the AWS partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, you can configure other AWS services to further analyze and act upon the event data collected in CloudTrail logs. For more information, see the following:

- Overview for creating a trail
- CloudTrail supported services and integrations
- Configuring Amazon SNS notifications for CloudTrail
- Receiving CloudTrail log files from multiple regions and Receiving CloudTrail log files from multiple accounts

All Discovery API actions are logged by CloudTrail and are documented in this API Reference. For example, calls to the SearchListings, GetSearchFacets, and GetListingView 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 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 in the AWS CloudTrail User Guide.

Understanding Discovery API log file entries

A trail is a configuration that enables delivery of events as log files to an Amazon 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 aren't an ordered stack trace of the public API calls, so they don't appear in any specific order.

The following example shows a CloudTrail log entry that demonstrates the SearchListings action.

```json
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "ABCDEFGHIJKLMNOP12345",
        "arn": "arn:aws:iam::123456789010:user/CloudTrailTestUser",
        "accountId": "123456789010",
        "accessKeyId": "ABCDEFGHIJKLMNOP1234",
        "userName": "CloudTrailTestUser",
        "sessionContext": {
            "sessionIssuer": {},
            "webIdFederationData": {},
            "attributes": {
```
Release notes for AWS Marketplace Discovery API

The release notes for AWS Marketplace Discovery API (Discovery API) provides details about the service's features, improvements, fixes, and announcements by release date.

Discovery API release notes for 2022

May 20, 2022

Published on May 20, 2022

Existing Discovery API customers can access the updated Discovery API documentation and SDK on the Amazon Simple Storage Service (Amazon S3) bucket that the Discovery API team shared with them previously. Customers can refer to the Change Log in the private documentation for more details.

Discovery API announces the following launch, and improvements:

Launch announcements

- Discovery API launched in two additional AWS Regions:
• US West (Oregon) – us-west-2
• Europe (Ireland) – eu-west-1

Discovery API Private SDK is now available in Java 2.x:
• For more information about how to use the AWS SDK for Java 2.x, see the [AWS SDK for Java 2.x Developer Guide](https://docs.aws.amazon.com/sdkforjava/latest/api/index.html).
• For more information about migration, see [migrating from version 1.x to 2.x of the AWS SDK for Java](https://docs.aws.amazon.com/sdkforjava/latest/api/migration.html) in the [AWS SDK for Java 2.x Developer Guide](https://docs.aws.amazon.com/sdkforjava/latest/api/index.html).
• For more information about changes between versions 1.11.x and 2.x of the AWS SDK for Java 2.x, see [1.11 to 2.x Changelog](https://github.com/aws/aws-sdk-java/releases) on the GitHub website.

**Improvements**

• Enhanced sorting functionality for the `SearchListings` API operation by introducing new options for:
  • `SortBy` – AVERAGE_CUSTOMER_RATING, CREATION_TIME, LAST_MODIFIED_TIME
  • `SortOrder` – ASCENDING
• SDK updated for all the existing languages with the latest AWS SDK artifacts.
• Documentation updated to include SDK usage section for all languages.
AWS Marketplace Agreements API

The AWS Marketplace Catalog API service provides a service to manage agreements.

Logging Agreements API calls using AWS CloudTrail

The Agreements API is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in AWS Marketplace. CloudTrail captures API calls for the Agreements API as events. The calls captured include calls from the AWS Marketplace website, console, and other interfaces leveraging the Agreements API, as well as direct code calls to Agreements API operations.

If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for the Agreements API. A trail enables CloudTrail to deliver log files to an Amazon S3 bucket. 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 the Agreements API, the IP address from which the request was made, who made the request, when it was made, and additional details.

For more information about CloudTrail, see the AWS CloudTrail User Guide.

Agreements API information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When activity occurs in the Agreements API, that 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 the Agreements API, create a trail. A trail enables CloudTrail to deliver log files to an Amazon S3 bucket. By default, when you create a trail in the console, the trail applies to all AWS Regions. The trail logs events from all Regions in the AWS partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, you can configure other AWS services to further analyze and act upon the event data collected in CloudTrail logs. For more information, see the following:

• Overview for creating a trail
• CloudTrail supported services and integrations
• Configuring Amazon SNS notifications for CloudTrail
• Receiving CloudTrail log files from multiple regions and Receiving CloudTrail log files from multiple accounts

The AcceptAgreementRequest and CancelAgreement Agreements API actions are logged by CloudTrail.

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 user credentials.
• Whether the request was made with temporary security credentials for a role or federated user.
Understanding Agreements API actions

The Agreements API is used to purchase software as a service (SaaS), server (including container), and professional services products on AWS Marketplace. It’s also used to manage resulting agreements or subscriptions on AWS Marketplace.

Note

Purchases of machine learning or AWS Data Exchange products won’t be logged by CloudTrail.

The AcceptAgreementRequest action is used when an AWS Identity and Access Management (IAM) user or role of an AWS account purchases an applicable product on AWS Marketplace. Similarly, the CancelAgreement action is used when an IAM user or role cancels their agreement or subscription. By monitoring CloudTrail logs in the Agreements API, buyers can monitor the most important purchase-related actions happening in their AWS account on AWS Marketplace.

Buyers can also identify the Agreement ID of the agreement from the CloudTrail log. For more information about the agreement, choose the Manage subscriptions tab in the AWS Marketplace console, where the Agreement ID is provided in the Details view. The Agreement ID can be found in responseElements for the AcceptAgreementRequest API action and in requestParameters for the CancelAgreement API action.

Understanding Agreements API log file entries

A trail is a configuration that enables delivery of events as log files to an Amazon 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 aren't an ordered stack trace of the public API calls, so they don't display in a specific order.

The following example shows a CloudTrail log entry that demonstrates the AcceptAgreementRequest action.

```json
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "Unknown",
        "principalId": "ABCDEFGHIJKLMNOP12345",
        "arn": "arn:aws:iam::123456789010:user/CloudTrailTestUser",
        "accountId": "123456789010",
        "accessKeyId": "ABCDEFGHIJKLMNOP123"
    },
    "eventTime": "2023-08-11T17:13:50Z",
    "eventSource": "agreement-marketplace.amazonaws.com",
    "eventName": "AcceptAgreementRequest",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "127.0.0.1",
    "userAgent": "Coral/Netty4",
    "requestParameters": {
        "agreementRequestId": "ar-6xbrddjzym594imkrrezrn5wa"
    },
    "responseElements": {
        "agreementId": "agmt-1lnrq6riwpg2tczhv378zknlc"
    },
    "requestID": "fEXAMPLE-cb3e-4e21-86fd-6b3EXAMPLEEd1",
    "eventID": "7EXAMPLE-97d6-4159-91e3-01aEXAMPLE48"
}
```
The following example shows a CloudTrail log entry that demonstrates the CancelAgreement action.

```json
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "Unknown",
    "principalId": "ABCDEFGHIJKLMNOPQRSTUVWXYZ012345",
    "arn": "arn:aws:iam::123456789010:user/CloudTrailTestUser",
    "accountId": "123456789010",
    "accessKeyId": "ABCDEFGHIJKLMNOPQRSTUVWXYZ01234"
  },
  "eventTime": "2023-08-14T03:11:42Z",
  "eventSource": "agreement-marketplace.amazonaws.com",
  "eventName": "CancelAgreement",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "127.0.0.1",
  "userAgent": "Coral/Netty4",
  "requestParameters": {
    "agreementId": "agmt-enitbfqjebmzumrzc032t"
  },
  "responseElements": null,
  "requestID": "fEXAMPLE-dcb3-4e2e-86fd-6b3EXAMPLEd1",
  "eventID": "7EXAMPLE-97d6-4139-91e5-01aEXAMPLE48",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "123456789010",
  "eventCategory": "Management"
}
```
Actions

The following actions are supported:

- `CancelChangeSet (p. 88)`
- `DeleteResourcePolicy (p. 91)`
- `DescribeChangeSet (p. 93)`
- `DescribeEntity (p. 97)`
- `GetResourcePolicy (p. 100)`
- `ListChangeSets (p. 102)`
- `ListEntities (p. 106)`
- `ListTagsForResource (p. 110)`
- `PutResourcePolicy (p. 113)`
- `StartChangeSet (p. 115)`
- `TagResource (p. 120)`
- `UntagResource (p. 123)`
CancelChangeSet

Used to cancel an open change request. Must be sent before the status of the request changes to APPLYING, the final stage of completing your change request. You can describe a change during the 60-day request history retention period for API calls.

Request Syntax

PATCH /CancelChangeSet?catalog=Catalog&changeSetId=ChangeSetId HTTP/1.1

URI Request Parameters

The request uses the following URI parameters.

Catalog (p. 88)
- Required. The catalog related to the request. Fixed value: AWSMarketplace.
- Length Constraints: Minimum length of 1. Maximum length of 64.
- Pattern: ^[a-zA-Z]+$
- Required: Yes

ChangeSetId (p. 88)
- Required. The unique identifier of the StartChangeSet request that you want to cancel.
- Pattern: ^\[\w-]+$
- Required: Yes

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
Content-type: application/json

```
{
  "ChangeSetArn": "string",
  "ChangeSetId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.
**ChangeSetArn (p. 88)**

The ARN associated with the change set referenced in this request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `^[a-zA-Z0-9:*/-]+$`

**ChangeSetId (p. 88)**

The unique identifier for the change set referenced in this request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[\w\-]+$`

**Errors**

For information about the errors that are common to all actions, see [Common Errors (p. 143)](#).

**AccessDeniedException**

Access is denied.

HTTP status code: 403

HTTP Status Code: 403

**InternalServiceException**

There was an internal service exception.

HTTP status code: 500

HTTP Status Code: 500

**ResourceInUseException**

The resource is currently in use.

HTTP Status Code: 423

**ResourceNotFoundException**

The specified resource wasn't found.

HTTP status code: 404

HTTP Status Code: 404

**ThrottlingException**

Too many requests.

HTTP status code: 429

HTTP Status Code: 429

**ValidationException**

An error occurred during validation.
HTTP status code: 422
HTTP Status Code: 422

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteResourcePolicy

Deletes a resource-based policy on an entity that is identified by its resource ARN.

Request Syntax

```
DELETE /DeleteResourcePolicy?resourceArn=ResourceArn  HTTP/1.1
```

URI Request Parameters

The request uses the following URI parameters.

ResourceArn (p. 91)

The Amazon Resource Name (ARN) of the entity resource that is associated with the resource policy.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: ^arn:[\w+=/,.@-]+:aws-marketplace:[\w+=/,.@-]*:[0-9]+:[\w+=,.@-]+(/[\w+=,.@-]+)*$

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 143).

AccessDeniedException

Access is denied.

HTTP status code: 403

InternalServiceException

There was an internal service exception.

HTTP status code: 500
HTTP Status Code: 500

**ResourceNotFoundException**

The specified resource wasn't found.

HTTP status code: 404

HTTP Status Code: 404

**ThrottlingException**

Too many requests.

HTTP status code: 429

HTTP Status Code: 429

**ValidationException**

An error occurred during validation.

HTTP status code: 422

HTTP Status Code: 422

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](http://awscli.amazonaws.com
- [AWS SDK for .NET](http://aws.amazon.com
- [AWS SDK for C++](http://aws.amazon.com
- [AWS SDK for Go](http://aws.amazon.com
- [AWS SDK for Java V2](http://aws.amazon.com
- [AWS SDK for JavaScript](http://aws.amazon.com
- [AWS SDK for PHP V3](http://aws.amazon.com
- [AWS SDK for Python](http://aws.amazon.com
- [AWS SDK for Ruby V3](http://aws.amazon.com


DescribeChangeSet

Provides information about a given change set.

Request Syntax

GET /DescribeChangeSet?catalog=Catalog&changeSetId=ChangeSetId HTTP/1.1

URI Request Parameters

The request uses the following URI parameters.

**Catalog (p. 93)**

Required. The catalog related to the request. Fixed value: AWSMarketplace

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: ^[a-zA-Z]+$

Required: Yes

**ChangeSetId (p. 93)**

Required. The unique identifier for the StartChangeSet request that you want to describe the details for.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: ^\[\w\-]+$

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
Content-type: application/json

{
   "ChangeSet": [
      {
         "ChangeName": "string",
         "ChangeType": "string",
         "Details": "string",
         "DetailsDocument": JSON value,
         "Entity": {
            "Identifier": "string",
            "Type": "string"
         },
         "ErrorDetailList": [
            {
               "ErrorCode": "string",
            }
         ]
      }
   ]
}
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**ChangeSet (p. 93)**

An array of ChangeSummary objects.

Type: Array of ChangeSummary (p. 131) objects

**ChangeSetArn (p. 93)**

The ARN associated with the unique identifier for the change set referenced in this request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: ^[a-zA-Z0-9\_\-]+$

**ChangeSetId (p. 93)**

Required. The unique identifier for the change set referenced in this request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: ^[\w\-]+$

**ChangeSetName (p. 93)**

The optional name provided in the StartChangeSet request. If you do not provide a name, one is set by default.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: ^[\w\s+=.:@-]+$

**EndTime (p. 93)**

The date and time, in ISO 8601 format (2018-02-27T13:45:22Z), the request transitioned to a terminal state. The change cannot transition to a different state. Null if the request is not in a terminal state.

**FailureCode (p. 93)**

**FailureDescription (p. 93)**

**StartTime (p. 93)**

**Status (p. 93)**
Type: String
Pattern: ^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|1[2][0-9])T(2[0-3]|0[1-9]|0)\:([0-5][\d]):([0-5][\d])Z$

**FailureCode (p. 93)**

Returned if the change set is in FAILED status. Can be either CLIENT_ERROR, which means that there are issues with the request (see the ErrorDetailList), or SERVER_FAULT, which means that there is a problem in the system, and you should retry your request.

Type: String

Valid Values: CLIENT_ERROR | SERVER_FAULT

**FailureDescription (p. 93)**

Returned if there is a failure on the change set, but that failure is not related to any of the changes in the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: ^(. )+$

**StartTime (p. 93)**

The date and time, in ISO 8601 format (2018-02-27T13:45:22Z), the request started.

Type: String


Pattern: ^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|1[2][0-9])T(2[0-3]|0[1-9]|0)\:([0-5][\d]):([0-5][\d])Z$

**Status (p. 93)**

The status of the change request.

Type: String

Valid Values: PREPARING | APPLYING | SUCCEEDED | CANCELLED | FAILED

**Errors**

For information about the errors that are common to all actions, see [Common Errors (p. 143)](#).

**AccessDeniedException**

Access is denied.

HTTP status code: 403

**InternalServiceException**

There was an internal service exception.

HTTP status code: 500
HTTP Status Code: 500

**ResourceNotFoundException**

The specified resource wasn't found.

HTTP status code: 404

HTTP Status Code: 404

**ThrottlingException**

Too many requests.

HTTP status code: 429

HTTP Status Code: 429

**ValidationException**

An error occurred during validation.

HTTP status code: 422

HTTP Status Code: 422

---

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)
DescribeEntity

Returns the metadata and content of the entity.

Request Syntax

GET /DescribeEntity?catalog=Catalog&entityId=EntityId HTTP/1.1

URI Request Parameters

The request uses the following URI parameters.

Catalog (p. 97)

Required. The catalog related to the request. Fixed value: AWSMarketplace
Length Constraints: Minimum length of 1. Maximum length of 64.
Pattern: ^[a-zA-Z]+$
Required: Yes

EntityId (p. 97)

Required. The unique ID of the entity to describe.
Length Constraints: Minimum length of 1. Maximum length of 255.
Pattern: ^[\w-]+$
Required: Yes

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
Content-type: application/json

{
  "Details": "string",
  "DetailsDocument": JSON value,
  "EntityArn": "string",
  "EntityIdentifier": "string",
  "EntityType": "string",
  "LastModifiedDate": "string"
}

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.
Details (p. 97)
This stringified JSON object includes the details of the entity.
Type: String
Pattern: ^\s*\{\s*\}\s*$

DetailsDocument (p. 97)
The JSON value of the details specific to the entity.
Type: JSON value

EntityArn (p. 97)
The ARN associated to the unique identifier for the entity referenced in this request.
Type: String
Length Constraints: Minimum length of 1. Maximum length of 2048.
Pattern: ^[a-zA-Z0-9:*/-]+$

EntityIdentifier (p. 97)
The identifier of the entity, in the format of EntityId@RevisionId.
Type: String
Length Constraints: Minimum length of 1. Maximum length of 255.
Pattern: ^[\w-@]+$

EntityType (p. 97)
The named type of the entity, in the format of EntityType@Version.
Type: String
Length Constraints: Minimum length of 1. Maximum length of 255.
Pattern: ^[a-zA-Z]+$

LastModifiedDate (p. 97)
Type: String
Pattern: ^([\d]{4})\-(1[0-2]|0[1-9])\-(3[01]|0[1-9]|12)[\d]T(2[0-3]|0[1-9]|0[1-9])\d\d\d\d\d\:([0-5]\d|5\d):([0-5]\d|5\d)Z$

Errors
For information about the errors that are common to all actions, see Common Errors (p. 143).

AccessDeniedException
Access is denied.
HTTP status code: 403
HTTP Status Code: 403

**InternalServiceException**

There was an internal service exception.

HTTP status code: 500
HTTP Status Code: 500

**ResourceNotFoundException**

The specified resource wasn't found.

HTTP status code: 404
HTTP Status Code: 404

**ResourceNotSupported Excepti**

Currently, the specified resource is not supported.

HTTP Status Code: 415

**ThrottlingException**

Too many requests.

HTTP status code: 429
HTTP Status Code: 429

**ValidationException**

An error occurred during validation.

HTTP status code: 422
HTTP Status Code: 422

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)
GetResourcePolicy

Gets a resource-based policy of an entity that is identified by its resource ARN.

Request Syntax

GET /GetResourcePolicy?resourceArn=ResourceArn HTTP/1.1

URI Request Parameters

The request uses the following URI parameters.

ResourceArn (p. 100)

The Amazon Resource Name (ARN) of the entity resource that is associated with the resource policy.

Length Constraints: Minimum length of 1. Maximum length of 255.


Required: Yes

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
Content-type: application/json

{  "Policy": "string"
}

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

Policy (p. 100)

The policy document to set; formatted in JSON.

Type: String


Pattern: ^[\u0009\u000A\u000D\u0020-\u00FF]+$
Errors

For information about the errors that are common to all actions, see Common Errors (p. 143).

**AccessDeniedException**

Access is denied.

HTTP status code: 403

HTTP Status Code: 403

**InternalServiceException**

There was an internal service exception.

HTTP status code: 500

HTTP Status Code: 500

**ResourceNotFoundException**

The specified resource wasn't found.

HTTP status code: 404

HTTP Status Code: 404

**ThrottlingException**

Too many requests.

HTTP status code: 429

HTTP Status Code: 429

**ValidationException**

An error occurred during validation.

HTTP status code: 422

HTTP Status Code: 422

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ListChangeSets

Returns the list of change sets owned by the account being used to make the call. You can filter this list by providing any combination of entityId, ChangeSetName, and status. If you provide more than one filter, the API operation applies a logical AND between the filters.

You can describe a change during the 60-day request history retention period for API calls.

Request Syntax

```
POST /ListChangeSets HTTP/1.1
Content-type: application/json

{
    "Catalog": "string",
    "FilterList": [
        {
            "Name": "string",
            "ValueList": [ "string" ]
        }
    ],
    "MaxResults": number,
    "NextToken": "string",
    "Sort": {
        "SortBy": "string",
        "SortOrder": "string"
    }
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

**Catalog (p. 102)**

The catalog related to the request. Fixed value: AWSMarketplace

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: ^[a-zA-Z]+$

Required: Yes

**FilterList (p. 102)**

An array of filter objects.

Type: Array of Filter (p. 137) objects

Array Members: Minimum number of 1 item. Maximum number of 8 items.

Required: No
MaxResults (p. 102)

The maximum number of results returned by a single call. This value must be provided in the next call to retrieve the next set of results. By default, this value is 20.

Type: Integer


Required: No

NextToken (p. 102)

The token value retrieved from a previous call to access the next page of results.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: ^[\w+=.:@\-\/]$

Required: No

Sort (p. 102)

An object that contains two attributes, SortBy and SortOrder.

Type: Sort (p. 139) object

Required: No

Response Syntax

HTTP/1.1 200
Content-type: application/json

```json
{
  "ChangeSetSummaryList": [
    {
      "ChangeSetArn": "string",
      "ChangeSetId": "string",
      "ChangeSetName": "string",
      "EndTime": "string",
      "EntityIdList": [ "string" ],
      "FailureCode": "string",
      "StartTime": "string",
      "Status": "string"
    }
  ],
  "NextToken": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ChangeSetSummaryList (p. 103)

Array of ChangeSetSummaryListItem objects.
Errors

For information about the errors that are common to all actions, see Common Errors (p. 143).

AccessDeniedException

Access is denied.

HTTP status code: 403
HTTP Status Code: 403

InternalServiceException

There was an internal service exception.

HTTP status code: 500
HTTP Status Code: 500

ThrottlingException

Too many requests.

HTTP status code: 429
HTTP Status Code: 429

ValidationException

An error occurred during validation.

HTTP status code: 422
HTTP Status Code: 422

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
See Also

- AWS SDK for Python
- AWS SDK for Ruby V3
ListEntities

Provides the list of entities of a given type.

Request Syntax

POST /ListEntities HTTP/1.1
Content-type: application/json

{
    "Catalog": "string",
    "EntityType": "string",
    "FilterList": [
        {
            "Name": "string",
            "ValueList": [ "string" ]
        }
    ],
    "MaxResults": number,
    "NextToken": "string",
    "OwnershipType": "string",
    "Sort": {
        "SortBy": "string",
        "SortOrder": "string"
    }
}

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

Catalog (p. 106)

The catalog related to the request. Fixed value: AWSMarketplace

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: ^[a-zA-Z]+$

Required: Yes

EntityType (p. 106)

The type of entities to retrieve.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: ^[a-zA-Z]+$

Required: Yes
FilterList (p. 106)

An array of filter objects. Each filter object contains two attributes, filterName and filterValues.

Type: Array of Filter (p. 137) objects

Array Members: Minimum number of 1 item. Maximum number of 8 items.

Required: No

MaxResults (p. 106)

Specifies the upper limit of the elements on a single page. If a value isn't provided, the default value is 20.

Type: Integer


Required: No

NextToken (p. 106)

The value of the next token, if it exists. Null if there are no more results.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: ^[\w+=.:@\-\/]$

Required: No

OwnershipType (p. 106)

Filters the returned set of entities based on their owner. The default is SELF. To list entities shared with you through AWS Resource Access Manager (AWS RAM), set to SHARED. Entities shared through the AWS Marketplace Catalog API PutResourcePolicy operation can't be discovered through the SHARED parameter.

Type: String

Valid Values: SELF | SHARED

Required: No

Sort (p. 106)

An object that contains two attributes, SortBy and SortOrder.

Type: Sort (p. 139) object

Required: No

Response Syntax

HTTP/1.1 200
Content-type: application/json

{
    "EntitySummaryList": [}
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**EntitySummaryList (p. 107)**

Array of EntitySummary object.

Type: Array of EntitySummary (p. 134) objects

**NextToken (p. 107)**

The value of the next token if it exists. Null if there is no more result.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: ^\w+=.:@-\/]$

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 143).

**AccessDeniedException**

Access is denied.

HTTP status code: 403

HTTP Status Code: 403

**InternalServiceException**

There was an internal service exception.

HTTP status code: 500

HTTP Status Code: 500

**ResourceNotFoundException**

The specified resource wasn't found.

HTTP status code: 404

HTTP Status Code: 404
ThrottlingException

Too many requests.

HTTP status code: 429
HTTP Status Code: 429

ValidationException

An error occurred during validation.

HTTP status code: 422
HTTP Status Code: 422

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)
ListTagsForResource

Lists all tags that have been added to a resource (either an entity or change set).

Request Syntax

```
POST /ListTagsForResource HTTP/1.1
Content-type: application/json
{
    "ResourceArn": "string"
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

**ResourceArn (p. 110)**

Required. The Amazon Resource Name (ARN) associated with the resource you want to list tags on.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: ^arn:[\w+=/,.@-]+:aws-marketplace:[\w+=/,.@-]*:[0-9]+:[\w+=,.@-]*(/ [\w+=,.@-]+)*$  

Required: Yes

Response Syntax

```
HTTP/1.1 200
Content-type: application/json
{
    "ResourceArn": "string",
    "Tags": [
        {
            "Key": "string",
            "Value": "string"
        }
    ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.
ResourceArn (p. 110)

Required. The ARN associated with the resource you want to list tags on.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: ^arn:\[\w+=/,.@-]+:aws-marketplace:\[\w+=/,.@-]*:\[0-9]+:\[\w+=,.@-]+(\[/\w+=,.@-]+)*$

Tags (p. 110)

Required. A list of objects specifying each key name and value. Number of objects allowed: 1-50.

Type: Array of Tag (p. 140) objects

Array Members: Minimum number of 1 item. Maximum number of 200 items.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 143).

AccessDeniedException

Access is denied.

HTTP status code: 403

HTTP Status Code: 403

InternalServiceException

There was an internal service exception.

HTTP status code: 500

HTTP Status Code: 500

ResourceNotFoundException

The specified resource wasn't found.

HTTP status code: 404

HTTP Status Code: 404

ThrottlingException

Too many requests.

HTTP status code: 429

HTTP Status Code: 429

ValidationException

An error occurred during validation.

HTTP status code: 422

HTTP Status Code: 422
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
PutResourcePolicy

Attaches a resource-based policy to an entity. Examples of an entity include: AmiProduct and ContainerProduct.

Request Syntax

POST /PutResourcePolicy HTTP/1.1
Content-type: application/json

{
   "Policy": "string",
   "ResourceArn": "string"
}

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

Policy (p. 113)

The policy document to set; formatted in JSON.

Type: String


Pattern: ^[\u0009\u000A\u000D\u0020-\u00FF]+$

Required: Yes

ResourceArn (p. 113)

The Amazon Resource Name (ARN) of the entity resource you want to associate with a resource policy.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: ^arn:\[\w+=/,.@-]+:aws-marketplace:\[\w+=/,.@-]*:[0-9]+:\[\w+=/,.@-]+(/\[\w+=/,.@-]+)*$}

Required: Yes

Response Syntax

HTTP/1.1 200

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.
Errors

For information about the errors that are common to all actions, see Common Errors (p. 143).

AccessDeniedException

Access is denied.

HTTP status code: 403

HTTP Status Code: 403

InternalServiceException

There was an internal service exception.

HTTP status code: 500

HTTP Status Code: 500

ResourceNotFoundException

The specified resource wasn’t found.

HTTP status code: 404

HTTP Status Code: 404

ThrottlingException

Too many requests.

HTTP status code: 429

HTTP Status Code: 429

ValidationException

An error occurred during validation.

HTTP status code: 422

HTTP Status Code: 422

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
StartChangeSet

Allows you to request changes for your entities. Within a single ChangeSet, you can't start the same change type against the same entity multiple times. Additionally, when a ChangeSet is running, all the entities targeted by the different changes are locked until the change set has completed (either succeeded, cancelled, or failed). If you try to start a change set containing a change against an entity that is already locked, you will receive a ResourceInUseException error.

For example, you can't start the ChangeSet described in the example later in this topic because it contains two changes to run the same change type (AddRevisions) against the same entity (entity-id@1).

For more information about working with change sets, see Working with change sets. For information about change types for single-AMI products, see Working with single-AMI products. Also, for more information about change types available for container-based products, see Working with container products.

Request Syntax

POST /StartChangeSet HTTP/1.1
Content-type: application/json

```
{
    "Catalog": "string",
    "ChangeSet": [
        {
            "ChangeName": "string",
            "ChangeType": "string",
            "Details": "string",
            "DetailsDocument": JSON value,
            "Entity": {
                "Identifier": "string",
                "Type": "string"
            },
            "EntityTags": [
                {
                    "Key": "string",
                    "Value": "string"
                }
            ],
            "ChangeSetName": "string",
            "ChangeSetTags": [
                {
                    "Key": "string",
                    "Value": "string"
                }
            ],
            "ClientRequestToken": "string"
        }
    ]
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.
Catalog (p. 115)

The catalog related to the request. Fixed value: AWSMarketplace

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: ^[a-zA-Z]+$

Required: Yes

ChangeSet (p. 115)

Array of change object.

Type: Array of Change (p. 127) objects

Array Members: Minimum number of 1 item. Maximum number of 20 items.

Required: Yes

ChangeSetName (p. 115)

Optional case sensitive string of up to 100 ASCII characters. The change set name can be used to filter the list of change sets.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: ^[\w\s+=.:@-]+$

Required: No

ChangeSetTags (p. 115)

A list of objects specifying each key name and value for the ChangeSetTags property.

Type: Array of Tag (p. 140) objects

Array Members: Minimum number of 1 item. Maximum number of 200 items.

Required: No

ClientRequestToken (p. 115)

A unique token to identify the request to ensure idempotency.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: ^[^!-~]+$

Required: No

Response Syntax

HTTP/1.1 200
Content-type: application/json
Response Elements

If the action is successful, the service sends back an HTTP 200 response. The following data is returned in JSON format by the service.

**ChangeSetArn (p. 116)**

The ARN associated to the unique identifier generated for the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `^[a-zA-Z0-9:*/-]+$`

**ChangeSetId (p. 116)**

Unique identifier generated for the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[\w\-]+$`

Errors

For information about the errors that are common to all actions, see [Common Errors (p. 143)].

**AccessDeniedException**

Access is denied.

HTTP status code: 403

HTTP Status Code: 403

**InternalServiceException**

There was an internal service exception.

HTTP status code: 500

HTTP Status Code: 500

**ResourceInUseException**

The resource is currently in use.

HTTP Status Code: 423

**ResourceNotFoundException**

The specified resource wasn't found.

HTTP status code: 404
HTTP Status Code: 404

**ServiceQuotaExceededException**

The maximum number of open requests per account has been exceeded.

HTTP Status Code: 402

**ThrottlingException**

Too many requests.

HTTP status code: 429

HTTP Status Code: 429

**ValidationException**

An error occurred during validation.

HTTP status code: 422

HTTP Status Code: 422

Examples

Example

You can't start this ChangeSet because it contains two changes to run the same change type (AddRevisions) against the same entity (entity-id@1).

```json
{
  "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\"] }"
    },
    {
      "ChangeType": "AddRevisions",
      "Entity": {
        "Identifier": "entity-id@1",
        "Type": "DataProduct@1.0"
      },
      "Details": "{\"DataSetArn\": \"data-set-arn\", \"RevisionArns\": [\"revision-arn\"] }"
    }
  ]
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java V2
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
TagResource

Tags a resource (either an entity or change set).

Request Syntax

POST /TagResource HTTP/1.1
Content-type: application/json

{
   "ResourceArn": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

ResourceArn (p. 120)

Required. The Amazon Resource Name (ARN) associated with the resource you want to tag.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: ^arn:([^\w+=/,.@-]+:aws-marketplace:[^\w+=/,.@-]*[^0-9]+[^\w+=/,.@-]+(/[^\w+=/,.@-]+)+$.

Required: Yes

Tags (p. 120)

Required. A list of objects specifying each key name and value. Number of objects allowed: 1-50.

Type: Array of Tag (p. 140) objects

Array Members: Minimum number of 1 item. Maximum number of 200 items.

Required: Yes

Response Syntax

HTTP/1.1 200
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 143).

AccessDeniedException

Access is denied.
HTTP status code: 403
HTTP Status Code: 403

InternalServiceException

There was an internal service exception.
HTTP status code: 500
HTTP Status Code: 500

ResourceNotFoundException

The specified resource wasn't found.
HTTP status code: 404
HTTP Status Code: 404

ThrottlingException

Too many requests.
HTTP status code: 429
HTTP Status Code: 429

ValidationException

An error occurred during validation.
HTTP status code: 422
HTTP Status Code: 422

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
See Also

- AWS SDK for Python
- AWS SDK for Ruby V3
UntagResource

Removes a tag or list of tags from a resource (either an entity or change set).

Request Syntax

```plaintext
POST /UntagResource HTTP/1.1
Content-type: application/json
{
    "ResourceArn": "string",
    "TagKeys": [ "string" ]
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

**ResourceArn (p. 123)**

Required. The Amazon Resource Name (ARN) associated with the resource you want to remove the tag from.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: ^arn:[\w+=/,.@-]+:aws-marketplace:[\w+=/,.@-]+:[0-9]+:[\w+=/,.@-]+(/\[\w+=/,.@-]+)*$

Required: Yes

**TagKeys (p. 123)**

Required. A list of key names of tags to be removed. Number of strings allowed: 0-256.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 200 items.


Pattern: ^([\p{L}\p{Z}\p{N}\_\.:\+/\-\@]+)*$

Required: Yes

Response Syntax

```
HTTP/1.1 200
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 143).

AccessDeniedException

Access is denied.

HTTP status code: 403
HTTP Status Code: 403

InternalServiceException

There was an internal service exception.

HTTP status code: 500
HTTP Status Code: 500

ResourceNotFoundException

The specified resource wasn’t found.

HTTP status code: 404
HTTP Status Code: 404

ThrottlingException

Too many requests.

HTTP status code: 429
HTTP Status Code: 429

ValidationException

An error occurred during validation.

HTTP status code: 422
HTTP Status Code: 422

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
See Also

- AWS SDK for Python
- AWS SDK for Ruby V3
Data Types

The AWS Marketplace Catalog Service API contains several data types that various actions use. This section describes each data type in detail.

Note
The order of each element in a data type structure is not guaranteed. Applications should not assume a particular order.

The following data types are supported:

- Change (p. 127)
- ChangeSetSummaryListItem (p. 129)
- ChangeSummary (p. 131)
- Entity (p. 133)
- EntitySummary (p. 134)
- ErrorDetail (p. 136)
- Filter (p. 137)
- Sort (p. 139)
- Tag (p. 140)
Change

An object that contains the ChangeType, Details, and Entity.

**Contents**

**ChangeType**

Change types are single string values that describe your intention for the change. Each change type is unique for each EntityType provided in the change's scope. For more information on change types available for single-AMI products, see Working with single-AMI products. Also, for more information about change types available for container-based products, see Working with container products.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[A-Z][\w]*$`

Required: Yes

**Entity**

The entity to be changed.

Type: Entity (p. 133) object

Required: Yes

**ChangeName**

Optional name for the change.

Type: String


Pattern: `^[a-zA-Z]`$`

Required: No

**Details**

This object contains details specific to the change type of the requested change. For more information about change types available for single-AMI products, see Working with single-AMI products. Also, for more information about change types available for container-based products, see Working with container products.

Type: String


Pattern: `^[\s]*\{[\s\S]*\}[\s]*$`

Required: No

**DetailsDocument**

Alternative field that accepts a JSON value instead of a string for ChangeType details. You can use either Details or DetailsDocument, but not both.
Type: JSON value
Required: No

**EntityTags**

The tags associated with the change.

Type: Array of [Tag](p. 140) objects

Array Members: Minimum number of 1 item. Maximum number of 200 items.
Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)
**ChangeSetSummaryListItem**

A summary of a change set returned in a list of change sets when the ListChangeSets action is called.

**Contents**

**ChangeSetArn**

The ARN associated with the unique identifier for the change set referenced in this request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `^[a-zA-Z0-9:*/-]+$`

Required: No

**ChangeSetId**

The unique identifier for a change set.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `^[\w\-]+$`

Required: No

**ChangeSetName**

The non-unique name for the change set.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `^[\w\s+=.:@-]+$`

Required: No

**EndTime**

The time, in ISO 8601 format (2018-02-27T13:45:22Z), when the change set was finished.

Type: String


Pattern: `^([\d]{4})-(1[0-2]|0[1-9])-(3[01]|0[1-9]|[12][\d])T(2[0-3]|0[1-9]|0[0-9])\d{0,2}$`

Required: No

**EntityIdList**

This object is a list of entity IDs (string) that are a part of a change set. The entity ID list is a maximum of 20 entities. It must contain at least one entity.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 255.
Pattern: ^[\w\-]+$  
Required: No

**FailureCode**

Returned if the change set is in FAILED status. Can be either CLIENT_ERROR, which means that there are issues with the request (see the ErrorDetailList of DescribeChangeSet), or SERVER_FAULT, which means that there is a problem in the system, and you should retry your request.

Type: String

Valid Values: CLIENT_ERROR | SERVER_FAULT

Required: No

**StartTime**

The time, in ISO 8601 format (2018-02-27T13:45:22Z), when the change set was started.

Type: String


Pattern: ^([\d]{4})-(1[0-2]|0[1-9])-(3[01]|0[1-9]|[12]\[\d\])T(2[0-3]|0[1] \[\d\]):([0-5][\d]):([0-5][\d])Z$

Required: No

**Status**

The current status of the change set.

Type: String

Valid Values: PREPARING | APPLYING | SUCCEEDED | CANCELLED | FAILED

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)
**ChangeSummary**

This object is a container for common summary information about the change. The summary doesn't contain the whole change structure.

**Contents**

**ChangeName**

Optional name for the change.

Type: String


Pattern: ^[a-zA-Z]$

Required: No

**ChangeType**

The type of the change.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: ^[A-Z][\w]*$

Required: No

**Details**

This object contains details specific to the change type of the requested change.

Type: String


Pattern: ^[\s]*\{[^\s]*\}[\s]*$

Required: No

**DetailsDocument**

The JSON value of the details specific to the change type of the requested change.

Type: JSON value

Required: No

**Entity**

The entity to be changed.

Type: [Entity](p. 133) object

Required: No

**ErrorDetailList**

An array of ErrorDetail objects associated with the change.

Type: Array of ErrorDetail (p. 136) objects
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
Entity

An entity contains data that describes your product, its supported features, and how it can be used or launched by your customer.

Contents

Type

- The type of entity.
- Type: String
- Pattern: \^[a-zA-Z]+$\n- Required: Yes

Identifier

- The identifier for the entity.
- Type: String
- Pattern: \^[\w-@]+$\n- Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
EntitySummary

This object is a container for common summary information about the entity. The summary doesn't contain the whole entity structure, but it does contain information common across all entities.

Contents

EntityArn
The ARN associated with the unique identifier for the entity.
Type: String
Length Constraints: Minimum length of 1. Maximum length of 2048.
Pattern: ^[a-zA-Z0-9:*/-]+$
Required: No

EntityId
The unique identifier for the entity.
Type: String
Length Constraints: Minimum length of 1. Maximum length of 255.
Pattern: ^[\w-]+$
Required: No

EntityType
The type of the entity.
Type: String
Length Constraints: Minimum length of 1. Maximum length of 255.
Pattern: ^[a-zA-Z]+$
Required: No

LastModifiedDate
The last time the entity was published, using ISO 8601 format (2018-02-27T13:45:22Z).
Type: String
Pattern: ^([\d]{4})-(1[0-2]|0[1-9])-(3[01]|0[1-9]|[12]\d)T(2[0-3]|0[1-9]|[01]\d)([0-5]\d):([0-5]\d)$
Required: No

Name
The name for the entity. This value is not unique. It is defined by the seller.
Type: String
Length Constraints: Minimum length of 1. Maximum length of 255.
Visibility

The visibility status of the entity to buyers. This value can be Public (everyone can view the entity), Limited (the entity is visible to limited accounts only), or Restricted (the entity was published and then unpublished and only existing buyers can view it).

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: ^[a-zA-Z]+$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
ErrorDetail

Details about the error.

Contents

ErrorCode

The error code that identifies the type of error.

Type: String


Pattern: ^[a-zA-Z_]+$

Required: No

ErrorMessage

The message for the error.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: ^( . )+$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
AWS Marketplace Catalog API Reference
Filter

Filter

A filter object, used to optionally filter results from calls to the ListEntities and ListChangeSets actions.

Contents

Name

For ListEntities, the supported value for this is an EntityId.

For ListChangeSets, the supported values are as follows:

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: ^[a-zA-Z]+$

Required: No

ValueList

ListEntities - This is a list of unique EntityIds.

ListChangeSets - The supported filter names and associated ValueLists is as follows:

• ChangeSetName - The supported ValueList is a list of non-unique ChangeSetNames. These are defined when you call the StartChangeSet action.

• Status - The supported ValueList is a list of statuses for all change set requests.

• EntityId - The supported ValueList is a list of unique EntityIds.

• BeforeStartTime - The supported ValueList is a list of all change sets that started before the filter value.

• AfterStartTime - The supported ValueList is a list of all change sets that started after the filter value.

• BeforeEndTime - The supported ValueList is a list of all change sets that ended before the filter value.

• AfterEndTime - The supported ValueList is a list of all change sets that ended after the filter value.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: ^(.)+$

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java V2
• AWS SDK for Ruby V3
Sort

An object that contains two attributes, SortBy and SortOrder.

Contents

SortBy

For ListEntities, supported attributes include LastModifiedDate (default) and EntityId. In addition to LastModifiedDate and EntityId, each EntityType might support additional fields.

For ListChangeSets, supported attributes include StartTime and EndTime.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: ^[a-zA-Z]+$

Required: No

SortOrder

The sorting order. Can be ASCENDING or DESCENDING. The default value is DESCENDING.

Type: String

Valid Values: ASCENDING | DESCENDING

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
Tag

A list of objects specifying each key name and value.

Contents

Key

The key associated with the tag.

Type: String


Pattern: ^\p{L}\p{Z}\p{N}_\:.+/\-@*$

Required: Yes

Value

The value associated with the tag.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 256.

Pattern: ^\p{L}\p{Z}\p{N}_\:.+/\-@*$

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
Common Parameters

The following list contains the parameters that all actions use for signing Signature Version 4 requests with a query string. Any action-specific parameters are listed in the topic for that action. For more information about Signature Version 4, see Signing AWS API requests in the IAM User Guide.

**Action**

The action to be performed.

Type: string

Required: Yes

**Version**

The API version that the request is written for, expressed in the format YYYY-MM-DD.

Type: string

Required: Yes

**X-Amz-Algorithm**

The hash algorithm that you used to create the request signature.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Valid Values: AWS4-HMAC-SHA256

Required: Conditional

**X-Amz-Credential**

The credential scope value, which is a string that includes your access key, the date, the region you are targeting, the service you are requesting, and a termination string ("aws4_request"). The value is expressed in the following format: access_key/YYYYMMDD/region/service/aws4_request.

For more information, see Create a signed AWS API request in the IAM User Guide.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

**X-Amz-Date**

The date that is used to create the signature. The format must be ISO 8601 basic format (YYYYMMDD'T'HHMMSS'Z'). For example, the following date time is a valid X-Amz-Date value: 20120325T120000Z.

Condition: X-Amz-Date is optional for all requests; it can be used to override the date used for signing requests. If the Date header is specified in the ISO 8601 basic format, X-Amz-Date is not required. When X-Amz-Date is used, it always overrides the value of the Date header. For more information, see Elements of an AWS API request signature in the IAM User Guide.
Type: string
Required: Conditional

**X-Amz-Security-Token**

The temporary security token that was obtained through a call to AWS Security Token Service (AWS STS). For a list of services that support temporary security credentials from AWS STS, see AWS services that work with IAM in the IAM User Guide.

Condition: If you're using temporary security credentials from AWS STS, you must include the security token.

Type: string
Required: Conditional

**X-Amz-Signature**

Specifies the hex-encoded signature that was calculated from the string to sign and the derived signing key.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string
Required: Conditional

**X-Amz-SignedHeaders**

Specifies all the HTTP headers that were included as part of the canonical request. For more information about specifying signed headers, see Create a signed AWS API request in the IAM User Guide.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string
Required: Conditional
Common Errors

This section lists the errors common to the API actions of all AWS services. For errors specific to an API action for this service, see the topic for that API action.

**AccessDeniedException**

You do not have sufficient access to perform this action.

HTTP Status Code: 400

**IncompleteSignature**

The request signature does not conform to AWS standards.

HTTP Status Code: 400

**InternalFailure**

The request processing has failed because of an unknown error, exception or failure.

HTTP Status Code: 500

**InvalidAction**

The action or operation requested is invalid. Verify that the action is typed correctly.

HTTP Status Code: 400

**InvalidClientTokenId**

The X.509 certificate or AWS access key ID provided does not exist in our records.

HTTP Status Code: 403

**NotAuthorized**

You do not have permission to perform this action.

HTTP Status Code: 400

**OptInRequired**

The AWS access key ID needs a subscription for the service.

HTTP Status Code: 403

**RequestExpired**

The request reached the service more than 15 minutes after the date stamp on the request or more than 15 minutes after the request expiration date (such as for pre-signed URLs), or the date stamp on the request is more than 15 minutes in the future.

HTTP Status Code: 400

**ServiceUnavailable**

The request has failed due to a temporary failure of the server.

HTTP Status Code: 503

**ThrottlingException**

The request was denied due to request throttling.
HTTP Status Code: 400

**Validation>Error**

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400
## Document history

The following table describes the documentation for this release of the *AWS Marketplace Catalog API Reference*.

<table>
<thead>
<tr>
<th>Change</th>
<th>Description</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AWS Marketplace Catalog API supports resource sharing (p. 145)</strong></td>
<td>The AWS Marketplace Catalog API integrates with AWS Resource Access Manager (AWS RAM) to enable resource sharing. See <a href="#">Working with AWS RAM to share resources</a>.</td>
<td>April 12, 2023</td>
</tr>
<tr>
<td><strong>AWS Marketplace Discovery API topic update (p. 145)</strong></td>
<td>The AWS Marketplace Discovery API now supports CloudTrail. See <a href="#">Logging AWS Marketplace Discovery API calls using AWS CloudTrail</a>.</td>
<td>December 15, 2022</td>
</tr>
<tr>
<td><strong>AWS Marketplace supports archiving private marketplace experiences (p. 145)</strong></td>
<td>Buyers can now archive and reactivate private marketplace experiences in AWS Marketplace. See <a href="#">Working with a private marketplace</a>.</td>
<td>December 12, 2022</td>
</tr>
<tr>
<td><strong>AWS Marketplace Private marketplace granular permissions (p. 145)</strong></td>
<td>Buyers now have more granular permissions to manage private marketplace experiences. See <a href="#">Working with a private marketplace</a>.</td>
<td>September 8, 2022</td>
</tr>
<tr>
<td><strong>AWS Marketplace Discovery API release notes (p. 145)</strong></td>
<td>Added <a href="#">Release notes</a> for the AWS Marketplace Discovery API.</td>
<td>May 20, 2022</td>
</tr>
<tr>
<td><strong>AWS Marketplace Discovery API topic update (p. 145)</strong></td>
<td>Documentation-only update to the <a href="#">AWS Marketplace Discovery API topic</a>.</td>
<td>January 14, 2022</td>
</tr>
<tr>
<td><strong>Support for Helm chart delivery options and QuickLaunch for container-based products (p. 145)</strong></td>
<td>Added documentation for adding or updating Helm chart delivery options in container-based product versions, including enabling QuickLaunch for buyers. See <a href="#">Working with container-based products</a>.</td>
<td>November 29, 2021</td>
</tr>
<tr>
<td><strong>Support for managing seller products (p. 145)</strong></td>
<td>Added the ability to manage AMI and container products programmatically. See <a href="#">Working with seller products</a>.</td>
<td>March 26, 2021</td>
</tr>
<tr>
<td><strong>Support for managing private marketplaces (p. 145)</strong></td>
<td>Added the ability to create and maintain private marketplaces for AWS Organizations programmatically. See <a href="#">Working with a private marketplace</a>.</td>
<td>December 3, 2020</td>
</tr>
<tr>
<td>The AWS Marketplace Discovery API is now available (p. 145)</td>
<td>The Discovery API provides programmatic access to find products in the AWS Marketplace. For details, see Discovery API.</td>
<td>September 30, 2020</td>
</tr>
<tr>
<td>-------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------------------------</td>
<td>--------------------</td>
</tr>
<tr>
<td>The AWS Marketplace Catalog API is now generally available (p. 145)</td>
<td>This service provides an API interface for approved providers to programatically access the self-service publishing capabilities on the AWS Marketplace Management Portal.</td>
<td>November 12, 2019</td>
</tr>
</tbody>
</table>