Amazon Elastic Container Registry Public

API Reference

API Version 2020-10-30

Amazon Elastic Container Registry Public: API Reference

Table of Contents

Welcome

Amazon Elastic Container Registry Public (Amazon ECR Public) is a managed container image registry service. Amazon ECR provides both public and private registries to host your container images. You can use the Docker CLI or your preferred client to push, pull, and manage images. Amazon ECR provides a secure, scalable, and reliable registry for your Docker or Open Container Initiative (OCI) images. Amazon ECR supports public repositories with this API. For information about the Amazon ECR API for private repositories, see Amazon Elastic Container Registry API Reference.

This document was last published on October 6, 2021.

Actions

The following actions are supported:

BatchCheckLayerAvailability

Checks the availability of one or more image layers that are within a repository in a public registry. When an image is pushed to a repository, each image layer is checked to verify if it has been uploaded before. If it has been uploaded, then the image layer is skipped.

Note
This operation is used by the Amazon ECR proxy and is not generally used by customers for pulling and pushing images. In most cases, you should use the `docker` CLI to pull, tag, and push images.

Request Syntax

```
{
   "layerDigests": [ "string" ],
   "registryId": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 92).

The request accepts the following data in JSON format.

layerDigests (p. 3)

The digests of the image layers to check.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Length Constraints: Minimum length of 0. Maximum length of 1000.

Required: Yes

registryId (p. 3)

The AWS account ID associated with the public registry that contains the image layers to check. If you do not specify a registry, the default public registry is assumed.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: No

repositoryName (p. 3)

The name of the repository that's associated with the image layers to check.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

Required: Yes

Response Syntax

```
{
   "failures": [
      {
         "failureCode": "string",
         "failureReason": "string",
         "layerDigest": "string"
      }
   ],
   "layers": [
      {
         "layerAvailability": "string",
         "layerDigest": "string",
         "layerSize": number,
         "mediaType": "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.

failures (p. 4)

Any failures associated with the call.

Type: Array of LayerFailure (p. 78) objects

layers (p. 4)

A list of image layer objects that correspond to the image layer references in the request.

Type: Array of Layer (p. 77) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 94).

InvalidParameterException

The specified parameter is invalid. Review the available parameters for the API request.

HTTP Status Code: 400

RegistryNotFoundException

The registry doesn't exist.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository can't be found. Check the spelling of the specified repository and ensure that you're performing operations on the correct registry.

HTTP Status Code: 400

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

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

BatchDeleteImage

Deletes a list of specified images that are within a repository in a public registry. Images are specified with either an `imageTag` or `imageDigest`.

You can remove a tag from an image by specifying the image's tag in your request. When you remove the last tag from an image, the image is deleted from your repository.

You can completely delete an image (and all of its tags) by specifying the digest of the image in your request.

Request Syntax

```
{
   "imageIds": [
      {
         "imageDigest": "string",
         "imageTag": "string"
      }
   ],
   "registryId": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 92).

The request accepts the following data in JSON format.

imageIds (p. 6)

A list of image ID references that correspond to images to delete. The format of the `imageIds` reference is `imageTag=tag` or `imageDigest=digest`.

Type: Array of ImageIdentifier (p. 75) objects

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Required: Yes

registryId (p. 6)

The AWS account ID that's associated with the registry that contains the image to delete. If you do not specify a registry, the default public registry is assumed.

Type: String

Pattern: `[0-9]{12}`

Required: No

repositoryName (p. 6)

The repository in a public registry that contains the image to delete.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

Required: Yes

Response Syntax

```
{
   "failures": [
      {
         "failureCode": "string",
         "failureReason": "string",
         "imageId": {
            "imageDigest": "string",
            "imageTag": "string"
         }
      }
   ],
   "imageIds": [
      {
         "imageDigest": "string",
         "imageTag": "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.

failures (p. 7)

Any failures associated with the call.

Type: Array of ImageFailure (p. 74) objects

imageIds (p. 7)

The image IDs of the deleted images.

Type: Array of ImageIdentifier (p. 75) objects

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 94).

InvalidParameterException

The specified parameter is invalid. Review the available parameters for the API request.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository can't be found. Check the spelling of the specified repository and ensure that you're performing operations on the correct registry.

HTTP Status Code: 400

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

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

CompleteLayerUpload

Informs Amazon ECR that the image layer upload is complete for a specified public registry, repository name, and upload ID. You can optionally provide a `sha256` digest of the image layer for data validation purposes.

When an image is pushed, the CompleteLayerUpload API is called once for each new image layer to verify that the upload is complete.

> **Note**
> This operation is used by the Amazon ECR proxy and is not generally used by customers for pulling and pushing images. In most cases, you should use the `docker` CLI to pull, tag, and push images.

Request Syntax

```
{
   "layerDigests": [ "string" ],
   "registryId": "string",
   "repositoryName": "string",
   "uploadId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 92).

The request accepts the following data in JSON format.

layerDigests (p. 9)

The `sha256` digest of the image layer.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Pattern: `[a-zA-Z0-9-_+.]+:[a-fA-F0-9]+`

Required: Yes

registryId (p. 9)

The AWS account ID associated with the registry where layers are uploaded. If you do not specify a registry, the default public registry is assumed.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: No

repositoryName (p. 9)

The name of the repository in a public registry to associate with the image layer.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

Required: Yes

uploadId (p. 9)

The upload ID from a previous InitiateLayerUpload (p. 43) operation to associate with the image layer.

Type: String

Pattern: `[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

Required: Yes

Response Syntax

```
{
   "layerDigest": "string",
   "registryId": "string",
   "repositoryName": "string",
   "uploadId": "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.

layerDigest (p. 10)

The `sha256` digest of the image layer.

Type: String

Pattern: `[a-zA-Z0-9-_+.]+:[a-fA-F0-9]+`

registryId (p. 10)

The public registry ID that's associated with the request.

Type: String

Pattern: `[0-9]{12}`

repositoryName (p. 10)

The repository name that's associated with the request.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

uploadId (p. 10)

The upload ID that's associated with the layer.

Type: String

Pattern: `[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 94).

EmptyUploadException

The specified layer upload doesn't contain any layer parts.

HTTP Status Code: 400

InvalidLayerException

The layer digest calculation performed by Amazon ECR when the image layer doesn't match the digest specified.

HTTP Status Code: 400

InvalidParameterException

The specified parameter is invalid. Review the available parameters for the API request.

HTTP Status Code: 400

LayerAlreadyExistsException

The image layer already exists in the associated repository.

HTTP Status Code: 400

LayerPartTooSmallException

Layer parts must be at least 5 MiB in size.

HTTP Status Code: 400

RegistryNotFoundException

The registry doesn't exist.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository can't be found. Check the spelling of the specified repository and ensure that you're performing operations on the correct registry.

HTTP Status Code: 400

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

UnsupportedCommandException

The action isn't supported in this Region.

HTTP Status Code: 400

UploadNotFoundException

The upload can't be found, or the specified upload ID isn't valid for this repository.

HTTP Status Code: 400

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

CreateRepository

Creates a repository in a public registry. For more information, see Amazon ECR repositories in the *Amazon Elastic Container Registry User Guide*.

Request Syntax

```
{
   "catalogData": {
      "aboutText": "string",
      "architectures": [ "string" ],
      "description": "string",
      "logoImageBlob": blob,
      "operatingSystems": [ "string" ],
      "usageText": "string"
   },
   "repositoryName": "string",
   "tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 92).

The request accepts the following data in JSON format.

catalogData (p. 13)

The details about the repository that are publicly visible in the Amazon ECR Public Gallery.

Type: RepositoryCatalogDataInput (p. 89) object

Required: No

repositoryName (p. 13)

The name to use for the repository. This appears publicly in the Amazon ECR Public Gallery. The repository name can be specified on its own (for example `nginx-web-app`) or prepended with a namespace to group the repository into a category (for example `project-a/nginx-web-app`).

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

Required: Yes

tags (p. 13)

The metadata that you apply to each repository to help categorize and organize your repositories. Each tag consists of a key and an optional value. You define both of them. Tag keys can have a maximum character length of 128 characters, and tag values can have a maximum length of 256 characters.

Type: Array of Tag (p. 91) objects

Array Members: Minimum number of 0 items. Maximum number of 200 items.

Required: No

Response Syntax

```
{
   "catalogData": {
      "aboutText": "string",
      "architectures": [ "string" ],
      "description": "string",
      "logoUrl": "string",
      "marketplaceCertified": boolean,
      "operatingSystems": [ "string" ],
      "usageText": "string"
   },
   "repository": {
      "createdAt": number,
      "registryId": "string",
      "repositoryArn": "string",
      "repositoryName": "string",
      "repositoryUri": "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.

catalogData (p. 14)

The catalog data for a repository. This data is publicly visible in the Amazon ECR Public Gallery.

Type: RepositoryCatalogData (p. 87) object

repository (p. 14)

The repository that was created.

Type: Repository (p. 85) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 94).

InvalidParameterException

The specified parameter is invalid. Review the available parameters for the API request.

HTTP Status Code: 400

InvalidTagParameterException

An invalid parameter has been specified. Tag keys can have a maximum character length of 128 characters, and tag values can have a maximum length of 256 characters.

HTTP Status Code: 400

LimitExceededException

The operation didn't succeed because it would have exceeded a service limit for your account. For more information, see Amazon ECR Service Quotas in the Amazon Elastic Container Registry User Guide.

HTTP Status Code: 400

RepositoryAlreadyExistsException

The specified repository already exists in the specified registry.

HTTP Status Code: 400

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

TooManyTagsException

The list of tags on the repository is over the limit. The maximum number of tags that can be applied to a repository is 50.

HTTP Status Code: 400

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

DeleteRepository

Deletes a repository in a public registry. If the repository contains images, you must either manually delete all images in the repository or use the `force` option. This option deletes all images on your behalf before deleting the repository.

Request Syntax

```
{
   "force": boolean,
   "registryId": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 92).

The request accepts the following data in JSON format.

force (p. 16)

The force option can be used to delete a repository that contains images. If the force option is not used, the repository must be empty prior to deletion.

Type: Boolean

Required: No

registryId (p. 16)

The AWS account ID that's associated with the public registry that contains the repository to delete. If you do not specify a registry, the default public registry is assumed.

Type: String

Pattern: `[0-9]{12}`

Required: No

repositoryName (p. 16)

The name of the repository to delete.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

Required: Yes

Response Syntax

```
{
   "repository": {
```

```
        "createdAt": number,
        "registryId": "string",
        "repositoryArn": "string",
        "repositoryName": "string",
        "repositoryUri": "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.

repository (p. 16)

The repository that was deleted.

Type: Repository (p. 85) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 94).

InvalidParameterException

The specified parameter is invalid. Review the available parameters for the API request.

HTTP Status Code: 400

RepositoryNotEmptyException

The specified repository contains images. To delete a repository that contains images, you must force the deletion with the `force` parameter.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository can't be found. Check the spelling of the specified repository and ensure that you're performing operations on the correct registry.

HTTP Status Code: 400

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

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

DeleteRepositoryPolicy

Deletes the repository policy that's associated with the specified repository.

Request Syntax

```
{
   "registryId": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 92).

The request accepts the following data in JSON format.

registryId (p. 19)

The AWS account ID that's associated with the public registry that contains the repository policy to delete. If you do not specify a registry, the default public registry is assumed.

Type: String

Pattern: `[0-9]{12}`

Required: No

repositoryName (p. 19)

The name of the repository that's associated with the repository policy to delete.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

Required: Yes

Response Syntax

```
{
   "policyText": "string",
   "registryId": "string",
   "repositoryName": "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.

policyText (p. 19)

The JSON repository policy that was deleted from the repository.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 10240.

registryId (p. 19)

The registry ID that's associated with the request.

Type: String

Pattern: `[0-9]{12}`

repositoryName (p. 19)

The repository name that's associated with the request.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 94).

InvalidParameterException

The specified parameter is invalid. Review the available parameters for the API request.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository can't be found. Check the spelling of the specified repository and ensure that you're performing operations on the correct registry.

HTTP Status Code: 400

RepositoryPolicyNotFoundException

The specified repository and registry combination doesn't have an associated repository policy.

HTTP Status Code: 400

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

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

DescribeImages

Returns metadata that's related to the images in a repository in a public registry.

Note
Beginning with Docker version 1.9, the Docker client compresses image layers before pushing them to a V2 Docker registry. The output of the `docker images` command shows the uncompressed image size. Therefore, it might return a larger image size than the image sizes that are returned by DescribeImages (p. 22).

Request Syntax

```
{
   "imageIds": [
      {
         "imageDigest": "string",
         "imageTag": "string"
      }
   ],
   "maxResults": number,
   "nextToken": "string",
   "registryId": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 92).

The request accepts the following data in JSON format.

imageIds (p. 22)

The list of image IDs for the requested repository.

Type: Array of ImageIdentifier (p. 75) objects

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Required: No

maxResults (p. 22)

The maximum number of repository results that's returned by `DescribeImages` in paginated output. When this parameter is used, `DescribeImages` only returns `maxResults` results in a single page along with a `nextToken` response element. You can see the remaining results of the initial request by sending another `DescribeImages` request with the returned `nextToken` value. This value can be between 1 and 1000. If this parameter isn't used, then `DescribeImages` returns up to 100 results and a `nextToken` value, if applicable. If you specify images with `imageIds`, you can't use this option.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 1000.

Required: No

nextToken (p. 22)

The `nextToken` value that's returned from a previous paginated `DescribeImages` request where `maxResults` was used and the results exceeded the value of that parameter. Pagination continues from the end of the previous results that returned the `nextToken` value. If there are no more results to return, this value is `null`. If you specify images with `imageIds`, you can't use this option.

Type: String

Required: No

registryId (p. 22)

The AWS account ID that's associated with the public registry that contains the repository where images are described. If you do not specify a registry, the default public registry is assumed.

Type: String

Pattern: `[0-9]{12}`

Required: No

repositoryName (p. 22)

The repository that contains the images to describe.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

Required: Yes

Response Syntax

```
{
   "imageDetails": [
      {
         "artifactMediaType": "string",
         "imageDigest": "string",
         "imageManifestMediaType": "string",
         "imagePushedAt": number,
         "imageSizeInBytes": number,
         "imageTags": [ "string" ],
         "registryId": "string",
         "repositoryName": "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.

imageDetails (p. 23)

A list of ImageDetail (p. 72) objects that contain data about the image.

Type: Array of ImageDetail (p. 72) objects

nextToken (p. 23)

The `nextToken` value to include in a future `DescribeImages` request. When the results of a `DescribeImages` request exceed `maxResults`, you can use this value to retrieve the next page of results. If there are no more results to return, this value is `null`.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 94).

ImageNotFoundException

The image requested doesn't exist in the specified repository.

HTTP Status Code: 400

InvalidParameterException

The specified parameter is invalid. Review the available parameters for the API request.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository can't be found. Check the spelling of the specified repository and ensure that you're performing operations on the correct registry.

HTTP Status Code: 400

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

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

DescribeImageTags

Returns the image tag details for a repository in a public registry.

Request Syntax

```
{
   "maxResults": number,
   "nextToken": "string",
   "registryId": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 92).

The request accepts the following data in JSON format.

maxResults (p. 25)

The maximum number of repository results that's returned by `DescribeImageTags` in paginated output. When this parameter is used, `DescribeImageTags` only returns `maxResults` results in a single page along with a `nextToken` response element. You can see the remaining results of the initial request by sending another `DescribeImageTags` request with the returned `nextToken` value. This value can be between 1 and 1000. If this parameter isn't used, then `DescribeImageTags` returns up to 100 results and a `nextToken` value, if applicable. If you specify images with `imageIds`, you can't use this option.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 1000.

Required: No

nextToken (p. 25)

The `nextToken` value that's returned from a previous paginated `DescribeImageTags` request where `maxResults` was used and the results exceeded the value of that parameter. Pagination continues from the end of the previous results that returned the `nextToken` value. If there are no more results to return, this value is `null`. If you specify images with `imageIds`, you can't use this option.

Type: String

Required: No

registryId (p. 25)

The AWS account ID that's associated with the public registry that contains the repository where images are described. If you do not specify a registry, the default public registry is assumed.

Type: String

Pattern: `[0-9]{12}`

Required: No

repositoryName (p. 25)

The name of the repository that contains the image tag details to describe.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

Required: Yes

Response Syntax

```
{
   "imageTagDetails": [
      {
         "createdAt": number,
         "imageDetail": {
            "artifactMediaType": "string",
            "imageDigest": "string",
            "imageManifestMediaType": "string",
            "imagePushedAt": number,
            "imageSizeInBytes": number
         },
         "imageTag": "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.

imageTagDetails (p. 26)

The image tag details for the images in the requested repository.

Type: Array of ImageTagDetail (p. 76) objects

nextToken (p. 26)

The `nextToken` value to include in a future `DescribeImageTags` request. When the results of a `DescribeImageTags` request exceed `maxResults`, you can use this value to retrieve the next page of results. If there are no more results to return, this value is `null`.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 94).

InvalidParameterException

The specified parameter is invalid. Review the available parameters for the API request.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository can't be found. Check the spelling of the specified repository and ensure that you're performing operations on the correct registry.

HTTP Status Code: 400

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

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

DescribeRegistries

Returns details for a public registry.

Request Syntax

```
{
   "maxResults": number,
   "nextToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 92).

The request accepts the following data in JSON format.

maxResults (p. 28)

The maximum number of repository results that's returned by `DescribeRegistries` in paginated output. When this parameter is used, `DescribeRegistries` only returns `maxResults` results in a single page along with a `nextToken` response element. The remaining results of the initial request can be seen by sending another `DescribeRegistries` request with the returned `nextToken` value. This value can be between 1 and 1000. If this parameter isn't used, then `DescribeRegistries` returns up to 100 results and a `nextToken` value, if applicable.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 1000.

Required: No

nextToken (p. 28)

The `nextToken` value that's returned from a previous paginated `DescribeRegistries` request where `maxResults` was used and the results exceeded the value of that parameter. Pagination continues from the end of the previous results that returned the `nextToken` value. If there are no more results to return, this value is `null`.

Note
This token should be treated as an opaque identifier that is only used to retrieve the next items in a list and not for other programmatic purposes.

Type: String

Required: No

Response Syntax

```
{
   "nextToken": "string",
   "registries": [
      {
         "aliases": [
            {
```

```
                "defaultRegistryAlias": boolean,
                "name": "string",
                "primaryRegistryAlias": boolean,
                "status": "string"
            }
        ],
        "registryArn": "string",
        "registryId": "string",
        "registryUri": "string",
        "verified": boolean
    }
  ]
}
```

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.

nextToken (p. 28)

The `nextToken` value to include in a future `DescribeRepositories` request. If the results of a `DescribeRepositories` request exceed `maxResults`, you can use this value to retrieve the next page of results. If there are no more results, this value is `null`.

Type: String

registries (p. 28)

An object that contains the details for a public registry.

Type: Array of Registry (p. 81) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 94).

InvalidParameterException

The specified parameter is invalid. Review the available parameters for the API request.

HTTP Status Code: 400

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

UnsupportedCommandException

The action isn't supported in this Region.

HTTP Status Code: 400

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

DescribeRepositories

Describes repositories that are in a public registry.

Request Syntax

```
{
   "maxResults": number,
   "nextToken": "string",
   "registryId": "string",
   "repositoryNames": [ "string" ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 92).

The request accepts the following data in JSON format.

maxResults (p. 31)

The maximum number of repository results that's returned by `DescribeRepositories` in paginated output. When this parameter is used, `DescribeRepositories` only returns `maxResults` results in a single page along with a `nextToken` response element. You can see the remaining results of the initial request by sending another `DescribeRepositories` request with the returned `nextToken` value. This value can be between 1 and 1000. If this parameter isn't used, then `DescribeRepositories` returns up to 100 results and a `nextToken` value, if applicable. If you specify repositories with `repositoryNames`, you can't use this option.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 1000.

Required: No

nextToken (p. 31)

The `nextToken` value that's returned from a previous paginated `DescribeRepositories` request where `maxResults` was used and the results exceeded the value of that parameter. Pagination continues from the end of the previous results that returned the `nextToken` value. If there are no more results to return, this value is `null`. If you specify repositories with `repositoryNames`, you can't use this option.

Note
This token should be treated as an opaque identifier that is only used to retrieve the next items in a list and not for other programmatic purposes.

Type: String

Required: No

registryId (p. 31)

The AWS account ID that's associated with the registry that contains the repositories to be described. If you do not specify a registry, the default public registry is assumed.

Type: String

Pattern: `[0-9]{12}`

Required: No

repositoryNames (p. 31)

A list of repositories to describe. If this parameter is omitted, then all repositories in a registry are described.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

Required: No

Response Syntax

```
{
   "nextToken": "string",
   "repositories": [
      {
         "createdAt": number,
         "registryId": "string",
         "repositoryArn": "string",
         "repositoryName": "string",
         "repositoryUri": "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.

nextToken (p. 32)

The `nextToken` value to include in a future `DescribeRepositories` request. When the results of a `DescribeRepositories` request exceed `maxResults`, this value can be used to retrieve the next page of results. If there are no more results to return, this value is `null`.

Type: String

repositories (p. 32)

A list of repository objects corresponding to valid repositories.

Type: Array of Repository (p. 85) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 94).

InvalidParameterException

The specified parameter is invalid. Review the available parameters for the API request.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository can't be found. Check the spelling of the specified repository and ensure that you're performing operations on the correct registry.

HTTP Status Code: 400

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

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

GetAuthorizationToken

Retrieves an authorization token. An authorization token represents your IAM authentication credentials. You can use it to access any Amazon ECR registry that your IAM principal has access to. The authorization token is valid for 12 hours. This API requires the `ecr-public:GetAuthorizationToken` and `sts:GetServiceBearerToken` permissions.

Response Syntax

```
{
   "authorizationData": {
      "authorizationToken": "string",
      "expiresAt": number
   }
}
```

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.

authorizationData (p. 34)

An authorization token data object that corresponds to a public registry.

Type: AuthorizationData (p. 69) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 94).

InvalidParameterException

The specified parameter is invalid. Review the available parameters for the API request.

HTTP Status Code: 400

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

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

GetRegistryCatalogData

Retrieves catalog metadata for a public registry.

Response Syntax

```
{
   "registryCatalogData": {
      "displayName": "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.

registryCatalogData (p. 36)

The catalog metadata for the public registry.

Type: RegistryCatalogData (p. 84) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 94).

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

UnsupportedCommandException

The action isn't supported in this Region.

HTTP Status Code: 400

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

GetRepositoryCatalogData

Retrieve catalog metadata for a repository in a public registry. This metadata is displayed publicly in the Amazon ECR Public Gallery.

Request Syntax

```
{
   "registryId": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 92).

The request accepts the following data in JSON format.

registryId (p. 38)

The AWS account ID that's associated with the registry that contains the repositories to be described. If you do not specify a registry, the default public registry is assumed.

Type: String

Pattern: `[0-9]{12}`

Required: No

repositoryName (p. 38)

The name of the repository to retrieve the catalog metadata for.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

Required: Yes

Response Syntax

```
{
   "catalogData": {
      "aboutText": "string",
      "architectures": [ "string" ],
      "description": "string",
      "logoUrl": "string",
      "marketplaceCertified": boolean,
      "operatingSystems": [ "string" ],
      "usageText": "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.

catalogData (p. 38)

The catalog metadata for the repository.

Type: RepositoryCatalogData (p. 87) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 94).

InvalidParameterException

The specified parameter is invalid. Review the available parameters for the API request.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository can't be found. Check the spelling of the specified repository and ensure that you're performing operations on the correct registry.

HTTP Status Code: 400

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

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

GetRepositoryPolicy

Retrieves the repository policy for the specified repository.

Request Syntax

```
{
   "registryId": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 92).

The request accepts the following data in JSON format.

registryId (p. 40)

The AWS account ID that's associated with the public registry that contains the repository. If you do not specify a registry, the default public registry is assumed.

Type: String

Pattern: `[0-9]{12}`

Required: No

repositoryName (p. 40)

The name of the repository with the policy to retrieve.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

Required: Yes

Response Syntax

```
{
   "policyText": "string",
   "registryId": "string",
   "repositoryName": "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.

policyText (p. 40)

The repository policy text that's associated with the repository. The policy text will be in JSON format.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 10240.

registryId (p. 40)

The registry ID that's associated with the request.

Type: String

Pattern: `[0-9]{12}`

repositoryName (p. 40)

The repository name that's associated with the request.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 94).

InvalidParameterException

The specified parameter is invalid. Review the available parameters for the API request.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository can't be found. Check the spelling of the specified repository and ensure that you're performing operations on the correct registry.

HTTP Status Code: 400

RepositoryPolicyNotFoundException

The specified repository and registry combination doesn't have an associated repository policy.

HTTP Status Code: 400

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

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

InitiateLayerUpload

Notifies Amazon ECR that you intend to upload an image layer.

When an image is pushed, the InitiateLayerUpload API is called once for each image layer that hasn't already been uploaded. Whether an image layer uploads is determined by the BatchCheckLayerAvailability API action.

> **Note**
> This operation is used by the Amazon ECR proxy and is not generally used by customers for pulling and pushing images. In most cases, you should use the `docker` CLI to pull, tag, and push images.

Request Syntax

```
{
   "registryId": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 92).

The request accepts the following data in JSON format.

registryId (p. 43)

The AWS account ID that's associated with the registry to which you intend to upload layers. If you do not specify a registry, the default public registry is assumed.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: No

repositoryName (p. 43)

The name of the repository that you want to upload layers to.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

Required: Yes

Response Syntax

```
{
   "partSize": number,
   "uploadId": "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.

partSize (p. 43)

The size, in bytes, that Amazon ECR expects future layer part uploads to be.

Type: Long

Valid Range: Minimum value of 0.

uploadId (p. 43)

The upload ID for the layer upload. This parameter is passed to further UploadLayerPart (p. 64) and CompleteLayerUpload (p. 9) operations.

Type: String

Pattern: `[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 94).

InvalidParameterException

The specified parameter is invalid. Review the available parameters for the API request.

HTTP Status Code: 400

RegistryNotFoundException

The registry doesn't exist.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository can't be found. Check the spelling of the specified repository and ensure that you're performing operations on the correct registry.

HTTP Status Code: 400

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

UnsupportedCommandException

The action isn't supported in this Region.

HTTP Status Code: 400

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

List the tags for an Amazon ECR Public resource.

Request Syntax

```
{
   "resourceArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 92).

The request accepts the following data in JSON format.

resourceArn (p. 46)

The Amazon Resource Name (ARN) that identifies the resource to list the tags for. Currently, the supported resource is an Amazon ECR Public repository.

Type: String

Required: Yes

Response Syntax

```
{
   "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.

tags (p. 46)

The tags for the resource.

Type: Array of Tag (p. 91) objects

Array Members: Minimum number of 0 items. Maximum number of 200 items.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 94).

InvalidParameterException

The specified parameter is invalid. Review the available parameters for the API request.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository can't be found. Check the spelling of the specified repository and ensure that you're performing operations on the correct registry.

HTTP Status Code: 400

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

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

PutImage

Creates or updates the image manifest and tags that are associated with an image.

When an image is pushed and all new image layers have been uploaded, the PutImage API is called once to create or update the image manifest and the tags that are associated with the image.

Note
This operation is used by the Amazon ECR proxy and is not generally used by customers for pulling and pushing images. In most cases, you should use the `docker` CLI to pull, tag, and push images.

Request Syntax

```
{
   "imageDigest": "string",
   "imageManifest": "string",
   "imageManifestMediaType": "string",
   "imageTag": "string",
   "registryId": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 92).

The request accepts the following data in JSON format.

imageDigest (p. 48)

The image digest of the image manifest that corresponds to the image.

Type: String

Required: No

imageManifest (p. 48)

The image manifest that corresponds to the image to be uploaded.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4194304.

Required: Yes

imageManifestMediaType (p. 48)

The media type of the image manifest. If you push an image manifest that doesn't contain the `mediaType` field, you must specify the `imageManifestMediaType` in the request.

Type: String

Required: No

imageTag (p. 48)

The tag to associate with the image. This parameter is required for images that use the Docker Image Manifest V2 Schema 2 or Open Container Initiative (OCI) formats.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 300.

Required: No

registryId (p. 48)

The AWS account ID that's associated with the public registry that contains the repository where the image is put. If you do not specify a registry, the default public registry is assumed.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: No

repositoryName (p. 48)

The name of the repository where the image is put.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

Required: Yes

Response Syntax

```
{
   "image": {
      "imageId": {
         "imageDigest": "string",
         "imageTag": "string"
      },
      "imageManifest": "string",
      "imageManifestMediaType": "string",
      "registryId": "string",
      "repositoryName": "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.

image (p. 49)

Details of the image uploaded.

Type: Image (p. 70) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 94).

ImageAlreadyExistsException

The specified image has already been pushed, and there were no changes to the manifest or image tag after the last push.

HTTP Status Code: 400

ImageDigestDoesNotMatchException

The specified image digest doesn't match the digest that Amazon ECR calculated for the image.

HTTP Status Code: 400

ImageTagAlreadyExistsException

The specified image is tagged with a tag that already exists. The repository is configured for tag immutability.

HTTP Status Code: 400

InvalidParameterException

The specified parameter is invalid. Review the available parameters for the API request.

HTTP Status Code: 400

LayersNotFoundException

The specified layers can't be found, or the specified layer isn't valid for this repository.

HTTP Status Code: 400

LimitExceededException

The operation didn't succeed because it would have exceeded a service limit for your account. For more information, see Amazon ECR Service Quotas in the Amazon Elastic Container Registry User Guide.

HTTP Status Code: 400

ReferencedImagesNotFoundException

The manifest list is referencing an image that doesn't exist.

HTTP Status Code: 400

RegistryNotFoundException

The registry doesn't exist.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository can't be found. Check the spelling of the specified repository and ensure that you're performing operations on the correct registry.

HTTP Status Code: 400

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

UnsupportedCommandException

The action isn't supported in this Region.

HTTP Status Code: 400

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

PutRegistryCatalogData

Create or update the catalog data for a public registry.

Request Syntax

```
{
   "displayName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 92).

The request accepts the following data in JSON format.

displayName (p. 52)

The display name for a public registry. The display name is shown as the repository author in the Amazon ECR Public Gallery.

Note
The registry display name is only publicly visible in the Amazon ECR Public Gallery for verified accounts.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 100.

Required: No

Response Syntax

```
{
   "registryCatalogData": {
      "displayName": "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.

registryCatalogData (p. 52)

The catalog data for the public registry.

Type: RegistryCatalogData (p. 84) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 94).

InvalidParameterException

The specified parameter is invalid. Review the available parameters for the API request.

HTTP Status Code: 400

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

UnsupportedCommandException

The action isn't supported in this Region.

HTTP Status Code: 400

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

PutRepositoryCatalogData

Creates or updates the catalog data for a repository in a public registry.

Request Syntax

```
{
   "catalogData": {
      "aboutText": "string",
      "architectures": [ "string" ],
      "description": "string",
      "logoImageBlob": blob,
      "operatingSystems": [ "string" ],
      "usageText": "string"
   },
   "registryId": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 92).

The request accepts the following data in JSON format.

catalogData (p. 54)

An object containing the catalog data for a repository. This data is publicly visible in the Amazon ECR Public Gallery.

Type: RepositoryCatalogDataInput (p. 89) object

Required: Yes

registryId (p. 54)

The AWS account ID that's associated with the public registry the repository is in. If you do not specify a registry, the default public registry is assumed.

Type: String

Pattern: `[0-9]{12}`

Required: No

repositoryName (p. 54)

The name of the repository to create or update the catalog data for.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

Required: Yes

Response Syntax

```
{
   "catalogData": {
      "aboutText": "string",
      "architectures": [ "string" ],
      "description": "string",
      "logoUrl": "string",
      "marketplaceCertified": boolean,
      "operatingSystems": [ "string" ],
      "usageText": "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.

catalogData (p. 55)

The catalog data for the repository.

Type: RepositoryCatalogData (p. 87) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 94).

InvalidParameterException

The specified parameter is invalid. Review the available parameters for the API request.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository can't be found. Check the spelling of the specified repository and ensure that you're performing operations on the correct registry.

HTTP Status Code: 400

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

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

SetRepositoryPolicy

Applies a repository policy to the specified public repository to control access permissions. For more information, see Amazon ECR Repository Policies in the *Amazon Elastic Container Registry User Guide*.

Request Syntax

```
{
   "force": boolean,
   "policyText": "string",
   "registryId": "string",
   "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 92).

The request accepts the following data in JSON format.

force (p. 57)

If the policy that you want to set on a repository policy would prevent you from setting another policy in the future, you must force the SetRepositoryPolicy (p. 57) operation. This prevents accidental repository lockouts.

Type: Boolean

Required: No

policyText (p. 57)

The JSON repository policy text to apply to the repository. For more information, see Amazon ECR Repository Policies in the *Amazon Elastic Container Registry User Guide*.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 10240.

Required: Yes

registryId (p. 57)

The AWS account ID that's associated with the registry that contains the repository. If you do not specify a registry, the default public registry is assumed.

Type: String

Pattern: `[0-9]{12}`

Required: No

repositoryName (p. 57)

The name of the repository to receive the policy.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

Required: Yes

Response Syntax

```
{
   "policyText": "string",
   "registryId": "string",
   "repositoryName": "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.

policyText (p. 58)

The JSON repository policy text that's applied to the repository.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 10240.

registryId (p. 58)

The registry ID that's associated with the request.

Type: String

Pattern: `[0-9]{12}`

repositoryName (p. 58)

The repository name that's associated with the request.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 94).

InvalidParameterException

The specified parameter is invalid. Review the available parameters for the API request.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository can't be found. Check the spelling of the specified repository and ensure that you're performing operations on the correct registry.

HTTP Status Code: 400

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

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

Associates the specified tags to a resource with the specified `resourceArn`. If existing tags on a resource aren't specified in the request parameters, they aren't changed. When a resource is deleted, the tags associated with that resource are also deleted.

Request Syntax

```
{
   "resourceArn": "string",
   "tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 92).

The request accepts the following data in JSON format.

resourceArn (p. 60)

The Amazon Resource Name (ARN) of the resource to add tags to. Currently, the supported resource is an Amazon ECR Public repository.

Type: String

Required: Yes

tags (p. 60)

The tags to add to the resource. A tag is an array of key-value pairs. Tag keys can have a maximum character length of 128 characters, and tag values can have a maximum length of 256 characters.

Type: Array of Tag (p. 91) objects

Array Members: Minimum number of 0 items. Maximum number of 200 items.

Required: Yes

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. 94).

InvalidParameterException

The specified parameter is invalid. Review the available parameters for the API request.

HTTP Status Code: 400

InvalidTagParameterException

An invalid parameter has been specified. Tag keys can have a maximum character length of 128 characters, and tag values can have a maximum length of 256 characters.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository can't be found. Check the spelling of the specified repository and ensure that you're performing operations on the correct registry.

HTTP Status Code: 400

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

TooManyTagsException

The list of tags on the repository is over the limit. The maximum number of tags that can be applied to a repository is 50.

HTTP Status Code: 400

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

UntagResource

Deletes specified tags from a resource.

Request Syntax

```
{
   "resourceArn": "string",
   "tagKeys": [ "string" ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 92).

The request accepts the following data in JSON format.

resourceArn (p. 62)

The Amazon Resource Name (ARN) of the resource to delete tags from. Currently, the supported resource is an Amazon ECR Public repository.

Type: String

Required: Yes

tagKeys (p. 62)

The keys of the tags to be removed.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 200 items.

Length Constraints: Minimum length of 1. Maximum length of 128.

Required: Yes

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. 94).

InvalidParameterException

The specified parameter is invalid. Review the available parameters for the API request.

HTTP Status Code: 400

InvalidTagParameterException

An invalid parameter has been specified. Tag keys can have a maximum character length of 128 characters, and tag values can have a maximum length of 256 characters.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository can't be found. Check the spelling of the specified repository and ensure that you're performing operations on the correct registry.

HTTP Status Code: 400

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

TooManyTagsException

The list of tags on the repository is over the limit. The maximum number of tags that can be applied to a repository is 50.

HTTP Status Code: 400

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

UploadLayerPart

Uploads an image layer part to Amazon ECR.

When an image is pushed, each new image layer is uploaded in parts. The maximum size of each image layer part can be 20971520 bytes (about 20MB). The UploadLayerPart API is called once for each new image layer part.

Note
This operation is used by the Amazon ECR proxy and is not generally used by customers for pulling and pushing images. In most cases, you should use the `docker` CLI to pull, tag, and push images.

Request Syntax

```
{
   "layerPartBlob": blob,
   "partFirstByte": number,
   "partLastByte": number,
   "registryId": "string",
   "repositoryName": "string",
   "uploadId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 92).

The request accepts the following data in JSON format.

layerPartBlob (p. 64)

The base64-encoded layer part payload.

Type: Base64-encoded binary data object

Length Constraints: Minimum length of 0. Maximum length of 20971520.

Required: Yes

partFirstByte (p. 64)

The position of the first byte of the layer part witin the overall image layer.

Type: Long

Valid Range: Minimum value of 0.

Required: Yes

partLastByte (p. 64)

The position of the last byte of the layer part within the overall image layer.

Type: Long

Valid Range: Minimum value of 0.

Required: Yes

registryId (p. 64)

The AWS account ID that's associated with the registry that you're uploading layer parts to. If you do not specify a registry, the default public registry is assumed.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: No

repositoryName (p. 64)

The name of the repository that you're uploading layer parts to.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

Required: Yes

uploadId (p. 64)

The upload ID from a previous InitiateLayerUpload (p. 43) operation to associate with the layer part upload.

Type: String

Pattern: `[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

Required: Yes

Response Syntax

```
{
   "lastByteReceived": number,
   "registryId": "string",
   "repositoryName": "string",
   "uploadId": "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.

lastByteReceived (p. 65)

The integer value of the last byte that's received in the request.

Type: Long

Valid Range: Minimum value of 0.

registryId (p. 65)

The registry ID that's associated with the request.

Type: String

Pattern: `[0-9]{12}`

repositoryName (p. 65)

The repository name that's associated with the request.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

uploadId (p. 65)

The upload ID that's associated with the request.

Type: String

Pattern: `[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 94).

InvalidLayerPartException

The layer part size isn't valid, or the first byte specified isn't consecutive to the last byte of a previous layer part upload.

HTTP Status Code: 400

InvalidParameterException

The specified parameter is invalid. Review the available parameters for the API request.

HTTP Status Code: 400

LimitExceededException

The operation didn't succeed because it would have exceeded a service limit for your account. For more information, see Amazon ECR Service Quotas in the Amazon Elastic Container Registry User Guide.

HTTP Status Code: 400

RegistryNotFoundException

The registry doesn't exist.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository can't be found. Check the spelling of the specified repository and ensure that you're performing operations on the correct registry.

HTTP Status Code: 400

ServerException

These errors are usually caused by a server-side issue.

HTTP Status Code: 500

UnsupportedCommandException

The action isn't supported in this Region.

HTTP Status Code: 400

UploadNotFoundException

The upload can't be found, or the specified upload ID isn't valid for this repository.

HTTP Status Code: 400

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

Data Types

The Amazon Elastic Container Registry Public 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:

AuthorizationData

An authorization token data object that corresponds to a public registry.

Contents

authorizationToken

A base64-encoded string that contains authorization data for a public Amazon ECR registry. When the string is decoded, it's presented in the format `user:password` for public registry authentication using `docker login`.

Type: String

Pattern: `^\S+$`

Required: No

expiresAt

The Unix time in seconds and milliseconds when the authorization token expires. Authorization tokens are valid for 12 hours.

Type: Timestamp

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

Image

An object that represents an Amazon ECR image.

Contents

imageId

An object that contains the image tag and image digest associated with an image.

Type: ImageIdentifier (p. 75) object

Required: No

imageManifest

The image manifest that's associated with the image.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4194304.

Required: No

imageManifestMediaType

The manifest media type of the image.

Type: String

Required: No

registryId

The AWS account ID that's associated with the registry containing the image.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: No

repositoryName

The name of the repository that's associated with the image.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

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

ImageDetail

An object that describes an image that's returned by a DescribeImages (p. 22) operation.

Contents

artifactMediaType

The artifact media type of the image.

Type: String

Required: No

imageDigest

The `sha256` digest of the image manifest.

Type: String

Required: No

imageManifestMediaType

The media type of the image manifest.

Type: String

Required: No

imagePushedAt

The date and time, expressed in standard JavaScript date format, that the current image was pushed to the repository at.

Type: Timestamp

Required: No

imageSizeInBytes

The size, in bytes, of the image in the repository.

If the image is a manifest list, this is the max size of all manifests in the list.

Note
Beginning with Docker version 1.9, the Docker client compresses image layers before pushing them to a V2 Docker registry. The output of the `docker images` command shows the uncompressed image size, so it might return a larger image size than the image sizes that are returned by DescribeImages (p. 22).

Type: Long

Required: No

imageTags

The list of tags that's associated with this image.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 300.

Required: No

registryId

The AWS account ID that's associated with the public registry where this image belongs.

Type: String

Pattern: `[0-9]{12}`

Required: No

repositoryName

The name of the repository where this image belongs.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

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

ImageFailure

An object that represents an Amazon ECR image failure.

Contents

failureCode

The code that's associated with the failure.

Type: String

Valid Values: `InvalidImageDigest | InvalidImageTag | ImageTagDoesNotMatchDigest | ImageNotFound | MissingDigestAndTag | ImageReferencedByManifestList | KmsError`

Required: No

failureReason

The reason for the failure.

Type: String

Required: No

imageId

The image ID that's associated with the failure.

Type: ImageIdentifier (p. 75) object

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

ImageIdentifier

An object with identifying information for an Amazon ECR image.

Contents

imageDigest

The `sha256` digest of the image manifest.

Type: String

Required: No

imageTag

The tag that's used for the image.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 300.

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

ImageTagDetail

An object that represents the image tag details for an image.

Contents

createdAt

The time stamp that indicates when the image tag was created.

Type: Timestamp

Required: No

imageDetail

An object that describes the details of an image.

Type: ReferencedImageDetail (p. 79) object

Required: No

imageTag

The tag that's associated with the image.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 300.

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

Layer

An object that represents an Amazon ECR image layer.

Contents

layerAvailability

The availability status of the image layer.

Type: String

Valid Values: `AVAILABLE | UNAVAILABLE`

Required: No

layerDigest

The `sha256` digest of the image layer.

Type: String

Pattern: `[a-zA-Z0-9-_+.]+:[a-fA-F0-9]+`

Required: No

layerSize

The size, in bytes, of the image layer.

Type: Long

Required: No

mediaType

The media type of the layer, such as `application/vnd.docker.image.rootfs.diff.tar.gzip` or `application/vnd.oci.image.layer.v1.tar+gzip`.

Type: String

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

LayerFailure

An object that represents an Amazon ECR image layer failure.

Contents

failureCode

The failure code that's associated with the failure.

Type: String

Valid Values: `InvalidLayerDigest | MissingLayerDigest`

Required: No

failureReason

The reason for the failure.

Type: String

Required: No

layerDigest

The layer digest that's associated with the failure.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1000.

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

ReferencedImageDetail

An object that describes the image tag details that are returned by a DescribeImageTags (p. 25) action.

Contents

artifactMediaType

The artifact media type of the image.

Type: String

Required: No

imageDigest

The `sha256` digest of the image manifest.

Type: String

Required: No

imageManifestMediaType

The media type of the image manifest.

Type: String

Required: No

imagePushedAt

The date and time, expressed in standard JavaScript date format, which the current image tag was pushed to the repository at.

Type: Timestamp

Required: No

imageSizeInBytes

The size, in bytes, of the image in the repository.

If the image is a manifest list, this is the max size of all manifests in the list.

> **Note**
> Beginning with Docker version 1.9, the Docker client compresses image layers before pushing them to a V2 Docker registry. The output of the `docker images` command shows the uncompressed image size, so it might return a larger image size than the image sizes that are returned by DescribeImages (p. 22).

Type: Long

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

Registry

The details of a public registry.

Contents

aliases

An array of objects that represents the aliases for a public registry.

Type: Array of RegistryAlias (p. 82) objects

Required: Yes

registryArn

The Amazon Resource Name (ARN) of the public registry.

Type: String

Required: Yes

registryId

The AWS account ID that's associated with the registry. If you do not specify a registry, the default public registry is assumed.

Type: String

Pattern: `[0-9]{12}`

Required: Yes

registryUri

The URI of a public registry. The URI contains a universal prefix and the registry alias.

Type: String

Required: Yes

verified

Indicates whether the account is a verified AWS Marketplace vendor. If an account is verified, each public repository receives a verified account badge on the Amazon ECR Public Gallery.

Type: Boolean

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

RegistryAlias

An object representing the aliases for a public registry. A public registry is given an alias when it's created. However, a custom alias can be set using the Amazon ECR console. For more information, see Registries in the *Amazon Elastic Container Registry User Guide*.

Contents

defaultRegistryAlias

Indicates whether the registry alias is the default alias for the registry. When the first public repository is created, your public registry is assigned a default registry alias.

Type: Boolean

Required: Yes

name

The name of the registry alias.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 50.

Pattern: `[a-z][a-z0-9]+(?:[._-][a-z0-9]+)*`

Required: Yes

primaryRegistryAlias

Indicates whether the registry alias is the primary alias for the registry. If true, the alias is the primary registry alias and is displayed in both the repository URL and the image URI used in the `docker pull` commands on the Amazon ECR Public Gallery.

Note
A registry alias that isn't the primary registry alias can be used in the repository URI in a `docker pull` command.

Type: Boolean

Required: Yes

status

The status of the registry alias.

Type: String

Valid Values: `ACTIVE | PENDING | REJECTED`

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

RegistryCatalogData

The metadata for a public registry.

Contents

displayName

The display name for a public registry. This appears on the Amazon ECR Public Gallery.

Important
Only accounts that have the verified account badge can have a registry display name.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 100.

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

Repository

An object representing a repository.

Contents

createdAt

The date and time, in JavaScript date format, when the repository was created.

Type: Timestamp

Required: No

registryId

The AWS account ID that's associated with the public registry that contains the repository.

Type: String

Pattern: `[0-9]{12}`

Required: No

repositoryArn

The Amazon Resource Name (ARN) that identifies the repository. The ARN contains the `arn:aws:ecr` namespace, followed by the region of the repository, AWS account ID of the repository owner, repository namespace, and repository name. For example, `arn:aws:ecr:region:012345678910:repository/test`.

Type: String

Required: No

repositoryName

The name of the repository.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 205.

Pattern: `(?:[a-z0-9]+(?:[._-][a-z0-9]+)*/)*[a-z0-9]+(?:[._-][a-z0-9]+)*`

Required: No

repositoryUri

The URI for the repository. You can use this URI for container image `push` and `pull` operations.

Type: String

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

RepositoryCatalogData

The catalog data for a repository. This data is publicly visible in the Amazon ECR Public Gallery.

Contents

aboutText

The longform description of the contents of the repository. This text appears in the repository details on the Amazon ECR Public Gallery.

Type: String

Length Constraints: Maximum length of 10240.

Required: No

architectures

The architecture tags that are associated with the repository.

Note
Only supported operating system tags appear publicly in the Amazon ECR Public Gallery. For more information, see RepositoryCatalogDataInput (p. 89).

Type: Array of strings

Array Members: Maximum number of 50 items.

Length Constraints: Minimum length of 1. Maximum length of 50.

Required: No

description

The short description of the repository.

Type: String

Length Constraints: Maximum length of 1024.

Required: No

logoUrl

The URL that contains the logo that's associated with the repository.

Type: String

Length Constraints: Maximum length of 2048.

Required: No

marketplaceCertified

Indicates whether the repository is certified by AWS Marketplace.

Type: Boolean

Required: No

operatingSystems

The operating system tags that are associated with the repository.

Note
Only supported operating system tags appear publicly in the Amazon ECR Public Gallery. For more information, see RepositoryCatalogDataInput (p. 89).

Type: Array of strings

Array Members: Maximum number of 50 items.

Length Constraints: Minimum length of 1. Maximum length of 50.

Required: No

usageText

The longform usage details of the contents of the repository. The usage text provides context for users of the repository.

Type: String

Length Constraints: Maximum length of 10240.

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

RepositoryCatalogDataInput

An object that contains the catalog data for a repository. This data is publicly visible in the Amazon ECR Public Gallery.

Contents

aboutText

A detailed description of the contents of the repository. It's publicly visible in the Amazon ECR Public Gallery. The text must be in markdown format.

Type: String

Length Constraints: Maximum length of 10240.

Required: No

architectures

The system architecture that the images in the repository are compatible with. On the Amazon ECR Public Gallery, the following supported architectures appear as badges on the repository and are used as search filters.

Note
If an unsupported tag is added to your repository catalog data, it's associated with the repository and can be retrieved using the API but isn't discoverable in the Amazon ECR Public Gallery.

- `ARM`
- `ARM 64`
- `x86`
- `x86-64`

Type: Array of strings

Array Members: Maximum number of 50 items.

Length Constraints: Minimum length of 1. Maximum length of 50.

Required: No

description

A short description of the contents of the repository. This text appears in both the image details and also when searching for repositories on the Amazon ECR Public Gallery.

Type: String

Length Constraints: Maximum length of 1024.

Required: No

logoImageBlob

The base64-encoded repository logo payload.

Note
The repository logo is only publicly visible in the Amazon ECR Public Gallery for verified accounts.

Type: Base64-encoded binary data object

Length Constraints: Minimum length of 0. Maximum length of 512000.

Required: No

operatingSystems

The operating systems that the images in the repository are compatible with. On the Amazon ECR Public Gallery, the following supported operating systems appear as badges on the repository and are used as search filters.

Note
If an unsupported tag is added to your repository catalog data, it's associated with the repository and can be retrieved using the API but isn't discoverable in the Amazon ECR Public Gallery.

- `Linux`
- `Windows`

Type: Array of strings

Array Members: Maximum number of 50 items.

Length Constraints: Minimum length of 1. Maximum length of 50.

Required: No

usageText

Detailed information about how to use the contents of the repository. It's publicly visible in the Amazon ECR Public Gallery. The usage text provides context, support information, and additional usage details for users of the repository. The text must be in markdown format.

Type: String

Length Constraints: Maximum length of 10240.

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

The metadata that you apply to a resource to help you categorize and organize them. Each tag consists of a key and an optional value. You define both. Tag keys can have a maximum character length of 128 characters, and tag values can have a maximum length of 256 characters.

Contents

Key

One part of a key-value pair that make up a tag. A `key` is a general label that acts like a category for more specific tag values.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Required: No

Value

The optional part of a key-value pair that make up a tag. A `value` acts as a descriptor within a tag category (key).

Type: String

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

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

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 Signature Version 4 Signing Process in the *Amazon Web Services General Reference*.

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 Task 2: Create a String to Sign for Signature Version 4 in the *Amazon Web Services General Reference*.

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 Handling Dates in Signature Version 4 in the *Amazon Web Services General Reference*.

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 Security Token Service, go to AWS Services That Work with IAM in the *IAM User Guide*.

Condition: If you're using temporary security credentials from the AWS Security Token Service, 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 Task 1: Create a Canonical Request For Signature Version 4 in the *Amazon Web Services General Reference*.

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

InvalidParameterCombination

Parameters that must not be used together were used together.

HTTP Status Code: 400

InvalidParameterValue

An invalid or out-of-range value was supplied for the input parameter.

HTTP Status Code: 400

InvalidQueryParameter

The AWS query string is malformed or does not adhere to AWS standards.

HTTP Status Code: 400

MalformedQueryString

The query string contains a syntax error.

HTTP Status Code: 404

MissingAction

The request is missing an action or a required parameter.

HTTP Status Code: 400

MissingAuthenticationToken

The request must contain either a valid (registered) AWS access key ID or X.509 certificate.

HTTP Status Code: 403

MissingParameter

A required parameter for the specified action is not supplied.

HTTP Status Code: 400

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

ValidationError

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

HTTP Status Code: 400