
Amazon Elastic Container Registry

API Reference

API Version 2015-09-21



Amazon Elastic Container Registry: API Reference

Copyright © Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

Welcome	1
Actions	2
BatchCheckLayerAvailability	3
Request Syntax	3
Request Parameters	3
Response Syntax	4
Response Elements	4
Errors	4
Examples	5
See Also	6
BatchDeleteImage	7
Request Syntax	7
Request Parameters	7
Response Syntax	8
Response Elements	8
Errors	8
Examples	9
See Also	10
BatchGetImage	12
Request Syntax	12
Request Parameters	12
Response Syntax	13
Response Elements	13
Errors	14
Examples	14
See Also	15
CompleteLayerUpload	16
Request Syntax	16
Request Parameters	16
Response Syntax	17
Response Elements	17
Errors	18
See Also	19
CreateRepository	20
Request Syntax	20
Request Parameters	20
Response Syntax	21
Response Elements	21
Errors	22
Examples	22
See Also	23
DeleteLifecyclePolicy	25
Request Syntax	25
Request Parameters	25
Response Syntax	25
Response Elements	25
Errors	26
Examples	27
See Also	27
DeleteRegistryPolicy	29
Response Syntax	29
Response Elements	29
Errors	29
See Also	30

DeleteRepository	31
Request Syntax	31
Request Parameters	31
Response Syntax	31
Response Elements	32
Errors	32
Examples	33
See Also	33
DeleteRepositoryPolicy	35
Request Syntax	35
Request Parameters	35
Response Syntax	35
Response Elements	35
Errors	36
Examples	36
See Also	37
DescribeImageReplicationStatus	39
Request Syntax	39
Request Parameters	39
Response Syntax	39
Response Elements	40
Errors	40
See Also	41
DescribeImages	42
Request Syntax	42
Request Parameters	42
Response Syntax	43
Response Elements	44
Errors	44
Examples	45
See Also	46
DescribeImageScanFindings	47
Request Syntax	47
Request Parameters	47
Response Syntax	48
Response Elements	49
Errors	49
Examples	50
See Also	51
DescribeRegistry	53
Response Syntax	53
Response Elements	53
Errors	53
See Also	54
DescribeRepositories	55
Request Syntax	55
Request Parameters	55
Response Syntax	56
Response Elements	56
Errors	57
Examples	57
See Also	58
GetAuthorizationToken	59
Request Syntax	59
Request Parameters	59
Response Syntax	59
Response Elements	59

Errors	60
Examples	60
See Also	61
GetDownloadUrlForLayer	62
Request Syntax	62
Request Parameters	62
Response Syntax	63
Response Elements	63
Errors	63
See Also	64
GetLifecyclePolicy	65
Request Syntax	65
Request Parameters	65
Response Syntax	65
Response Elements	65
Errors	66
Examples	67
See Also	67
GetLifecyclePolicyPreview	69
Request Syntax	69
Request Parameters	69
Response Syntax	70
Response Elements	71
Errors	72
Examples	72
See Also	73
GetRegistryPolicy	74
Response Syntax	74
Response Elements	74
Errors	74
See Also	75
GetRepositoryPolicy	76
Request Syntax	76
Request Parameters	76
Response Syntax	76
Response Elements	76
Errors	77
Examples	77
See Also	78
InitiateLayerUpload	80
Request Syntax	80
Request Parameters	80
Response Syntax	80
Response Elements	81
Errors	81
See Also	81
ListImages	83
Request Syntax	83
Request Parameters	83
Response Syntax	84
Response Elements	84
Errors	85
Examples	85
See Also	86
ListTagsForResource	87
Request Syntax	87
Request Parameters	87

Response Syntax	87
Response Elements	87
Errors	87
Examples	88
See Also	89
PutImage	90
Request Syntax	90
Request Parameters	90
Response Syntax	91
Response Elements	91
Errors	91
Examples	93
See Also	94
PutImageScanningConfiguration	95
Request Syntax	95
Request Parameters	95
Response Syntax	95
Response Elements	96
Errors	96
Examples	97
See Also	97
PutImageTagMutability	99
Request Syntax	99
Request Parameters	99
Response Syntax	99
Response Elements	100
Errors	100
Examples	101
See Also	101
PutLifecyclePolicy	103
Request Syntax	103
Request Parameters	103
Response Syntax	103
Response Elements	104
Errors	104
Examples	104
See Also	105
PutRegistryPolicy	107
Request Syntax	107
Request Parameters	107
Response Syntax	107
Response Elements	107
Errors	108
See Also	108
PutReplicationConfiguration	109
Request Syntax	109
Request Parameters	109
Response Syntax	109
Response Elements	110
Errors	110
See Also	110
SetRepositoryPolicy	112
Request Syntax	112
Request Parameters	112
Response Syntax	113
Response Elements	113
Errors	113

Examples	114
See Also	115
StartImageScan	116
Request Syntax	116
Request Parameters	116
Response Syntax	116
Response Elements	117
Errors	117
Examples	118
See Also	119
StartLifecyclePolicyPreview	120
Request Syntax	120
Request Parameters	120
Response Syntax	120
Response Elements	121
Errors	121
Examples	122
See Also	123
TagResource	124
Request Syntax	124
Request Parameters	124
Response Elements	124
Errors	124
See Also	125
UntagResource	126
Request Syntax	126
Request Parameters	126
Response Elements	126
Errors	126
See Also	127
UploadLayerPart	128
Request Syntax	128
Request Parameters	128
Response Syntax	129
Response Elements	129
Errors	130
See Also	131
Data Types	132
Attribute	133
Contents	133
See Also	133
AuthorizationData	134
Contents	134
See Also	134
DescribeImagesFilter	135
Contents	135
See Also	135
EncryptionConfiguration	136
Contents	136
See Also	136
Image	138
Contents	138
See Also	138
ImageDetail	140
Contents	140
See Also	141
ImageFailure	142

Contents	142
See Also	142
ImageIdentifier	143
Contents	143
See Also	143
ImageReplicationStatus	144
Contents	144
See Also	144
ImageScanFinding	145
Contents	145
See Also	145
ImageScanFindings	146
Contents	146
See Also	146
ImageScanFindingsSummary	147
Contents	147
See Also	147
ImageScanningConfiguration	148
Contents	148
See Also	148
ImageScanStatus	149
Contents	149
See Also	149
Layer	150
Contents	150
See Also	150
LayerFailure	151
Contents	151
See Also	151
LifecyclePolicyPreviewFilter	152
Contents	152
See Also	152
LifecyclePolicyPreviewResult	153
Contents	153
See Also	153
LifecyclePolicyPreviewSummary	155
Contents	155
See Also	155
LifecyclePolicyRuleAction	156
Contents	156
See Also	156
ListImagesFilter	157
Contents	157
See Also	157
ReplicationConfiguration	158
Contents	158
See Also	158
ReplicationDestination	159
Contents	159
See Also	159
ReplicationRule	160
Contents	160
See Also	160
Repository	161
Contents	161
See Also	162
RepositoryFilter	163

Contents	163
See Also	163
Tag	164
Contents	164
See Also	164
Common Parameters	165
Common Errors	167

Welcome

Amazon Elastic Container Registry (Amazon ECR) is a managed container image registry service. Customers can use the familiar Docker CLI, or their 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 private repositories with resource-based permissions using IAM so that specific users or Amazon EC2 instances can access repositories and images.

Amazon ECR has service endpoints in each supported Region. For more information, see [Amazon ECR endpoints](#) in the *AWS General Reference*.

This document was last published on October 6, 2021.

Actions

The following actions are supported:

- [BatchCheckLayerAvailability](#) (p. 3)
- [BatchDeleteImage](#) (p. 7)
- [BatchGetImage](#) (p. 12)
- [CompleteLayerUpload](#) (p. 16)
- [CreateRepository](#) (p. 20)
- [DeleteLifecyclePolicy](#) (p. 25)
- [DeleteRegistryPolicy](#) (p. 29)
- [DeleteRepository](#) (p. 31)
- [DeleteRepositoryPolicy](#) (p. 35)
- [DescribeImageReplicationStatus](#) (p. 39)
- [DescribeImages](#) (p. 42)
- [DescribeImageScanFindings](#) (p. 47)
- [DescribeRegistry](#) (p. 53)
- [DescribeRepositories](#) (p. 55)
- [GetAuthorizationToken](#) (p. 59)
- [GetDownloadUrlForLayer](#) (p. 62)
- [GetLifecyclePolicy](#) (p. 65)
- [GetLifecyclePolicyPreview](#) (p. 69)
- [GetRegistryPolicy](#) (p. 74)
- [GetRepositoryPolicy](#) (p. 76)
- [InitiateLayerUpload](#) (p. 80)
- [ListImages](#) (p. 83)
- [ListTagsForResource](#) (p. 87)
- [PutImage](#) (p. 90)
- [PutImageScanningConfiguration](#) (p. 95)
- [PutImageTagMutability](#) (p. 99)
- [PutLifecyclePolicy](#) (p. 103)
- [PutRegistryPolicy](#) (p. 107)
- [PutReplicationConfiguration](#) (p. 109)
- [SetRepositoryPolicy](#) (p. 112)
- [StartImageScan](#) (p. 116)
- [StartLifecyclePolicyPreview](#) (p. 120)
- [TagResource](#) (p. 124)
- [UntagResource](#) (p. 126)
- [UploadLayerPart](#) (p. 128)

BatchCheckLayerAvailability

Checks the availability of one or more image layers in a repository.

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. 165).

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 registry that contains the image layers to check. If you do not specify a registry, the default registry is assumed.

Type: String

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

Required: No

repositoryName (p. 3)

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

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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. 151) objects

layers (p. 4)

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

Type: Array of [Layer](#) (p. 150) objects

Errors

For information about the errors that are common to all actions, see [Common Errors](#) (p. 167).

InvalidParameterException

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

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository could not be found. Check the spelling of the specified repository and ensure that you are 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

Examples

In the following example or examples, the Authorization header contents (AUTHPARAMS) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

This example checks the availability of an image layer in the amazonlinux repository.

Sample Request

```
POST / HTTP/1.1
Host: ecr.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 126
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.BatchCheckLayerAvailability
X-Amz-Date: 20161216T195733Zc
User-Agent: aws-cli/1.11.22 Python/2.7.12 Darwin/16.3.0 botocore/1.4.79
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{
  "layerDigests": [
    "sha256:8e3fa21c4cc40232e835a6761332d225c7af3235c5755f44ada2ed9d0e4ab7e8"
  ],
  "repositoryName": "amazonlinux"
}
```

Sample Response

```
HTTP/1.1 200 OK
Server: Server
Date: Fri, 16 Dec 2016 19:57:33 GMT
Content-Type: application/x-amz-json-1.1
Content-Length: 233
Connection: keep-alive
x-amzn-RequestId: e2422faf-c3c9-11e6-a3ee-63b3b5dcf3b9

{
  "failures": [],
  "layers": [
    {
      "layerAvailability": "AVAILABLE",
      "layerDigest":
        "sha256:8e3fa21c4cc40232e835a6761332d225c7af3235c5755f44ada2ed9d0e4ab7e8",
      "layerSize": 91768077,
      "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip"
    }
  ]
}
```

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 within a repository. 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 image's digest 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 \(p. 165\)](#).

The request accepts the following data in JSON format.

`imageIds` (p. 7)

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 \(p. 143\)](#) objects

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

Required: Yes

`registryId` (p. 7)

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

Type: String

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

Required: No

`repositoryName` (p. 7)

The repository that contains the image to delete.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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. 8)

Any failures associated with the call.

Type: Array of [ImageFailure](#) (p. 142) objects

imageIds (p. 8)

The image IDs of the deleted images.

Type: Array of [ImageIdentifier](#) (p. 143) 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. 167).

InvalidParameterException

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

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository could not be found. Check the spelling of the specified repository and ensure that you are 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

Examples

In the following example or examples, the Authorization header contents (AUTHPARAMS) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

This example deletes an image in the ubuntu repository with the imageTag value of xenial.

Sample Request

```
POST / HTTP/1.1
Host: ecr.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 66
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.BatchDeleteImage
X-Amz-Date: 20161216T193711Z
User-Agent: aws-cli/1.11.22 Python/2.7.12 Darwin/16.3.0 botocore/1.4.79
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{
  "repositoryName": "ubuntu",
  "imageIds": [
    {
      "imageTag": "xenial"
    }
  ]
}
```

Sample Response

```
HTTP/1.1 200 OK
Server: Server
Date: Fri, 16 Dec 2016 19:37:11 GMT
Content-Type: application/x-amz-json-1.1
Content-Length: 138
Connection: keep-alive
x-amzn-RequestId: 09cc7023-c3c7-11e6-8acf-61b7dd8abe56

{
  "failures": [],
  "imageIds": [
    {
      "imageDigest":
        "sha256:7a64bc9c8843b0a8c8b8a7e4715b7615e4e1b0d8ca3c7e7a76ec8250899c397a",
      "imageTag": "xenial"
    }
  ]
}
```

```
}  
]  
}
```

Example

This example deletes an image (and all of its tags) in the `ubuntu` repository with the `imageDigest` value of `sha256:7a64bc9c8843b0a8c8b8a7e4715b7615e4e1b0d8ca3c7e7a76ec8250899c397a`.

Sample Request

```
POST / HTTP/1.1  
Host: ecr.us-west-2.amazonaws.com  
Accept-Encoding: identity  
Content-Length: 134  
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.BatchDeleteImage  
X-Amz-Date: 20161216T194250Z  
User-Agent: aws-cli/1.11.22 Python/2.7.12 Darwin/16.3.0 botocore/1.4.79  
Content-Type: application/x-amz-json-1.1  
Authorization: AUTHPARAMS  
  
{  
  "repositoryName": "ubuntu",  
  "imageIds": [  
    {  
      "imageDigest":  
        "sha256:7a64bc9c8843b0a8c8b8a7e4715b7615e4e1b0d8ca3c7e7a76ec8250899c397a"  
    }  
  ]  
}
```

Sample Response

```
HTTP/1.1 200 OK  
Server: Server  
Date: Fri, 16 Dec 2016 19:42:50 GMT  
Content-Type: application/x-amz-json-1.1  
Content-Length: 248  
Connection: keep-alive  
x-amzn-RequestId: d441a9f6-c3c7-11e6-8acf-61b7dd8abe56  
  
{  
  "failures": [],  
  "imageIds": [  
    {  
      "imageDigest":  
        "sha256:7a64bc9c8843b0a8c8b8a7e4715b7615e4e1b0d8ca3c7e7a76ec8250899c397a",  
      "imageTag": "xenial"  
    },  
    {  
      "imageDigest":  
        "sha256:7a64bc9c8843b0a8c8b8a7e4715b7615e4e1b0d8ca3c7e7a76ec8250899c397a",  
      "imageTag": "latest"  
    }  
  ]  
}
```

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](#)

BatchGetImage

Gets detailed information for an image. Images are specified with either an `imageTag` or `imageDigest`.

When an image is pulled, the `BatchGetImage` API is called once to retrieve the image manifest.

Request Syntax

```
{
  "acceptedMediaTypes": [ "string" ],
  "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. 165).

The request accepts the following data in JSON format.

`acceptedMediaTypes` (p. 12)

The accepted media types for the request.

Valid values: `application/vnd.docker.distribution.manifest.v1+json` | `application/vnd.docker.distribution.manifest.v2+json` | `application/vnd.oci.image.manifest.v1+json`

Type: Array of strings

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

Required: No

`imageIds` (p. 12)

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

Type: Array of [ImageIdentifier](#) (p. 143) objects

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

Required: Yes

`registryId` (p. 12)

The AWS account ID associated with the registry that contains the images to describe. If you do not specify a registry, the default registry is assumed.

Type: String

Pattern: [0-9]{1,2}

Required: No

repositoryName (p. 12)

The repository that contains the images to describe.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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"
      }
    }
  ],
  "images": [
    {
      "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.

failures (p. 13)

Any failures associated with the call.

Type: Array of [ImageFailure](#) (p. 142) objects

images (p. 13)

A list of image objects corresponding to the image references in the request.

Type: Array of [Image](#) (p. 138) objects

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 167\)](#).

InvalidParameterException

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

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository could not be found. Check the spelling of the specified repository and ensure that you are 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

Examples

In the following example or examples, the Authorization header contents (AUTHPARAMS) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

This example describes an image in the amazonlinux repository with the imageTag value of latest.

Sample Request

```
POST / HTTP/1.1
Host: ecr.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 71
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.BatchGetImage
X-Amz-Date: 20161216T195356Z
User-Agent: aws-cli/1.11.22 Python/2.7.12 Darwin/16.3.0 botocore/1.4.79
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{
  "repositoryName": "amazonlinux",
  "imageIds": [
    {
      "imageTag": "latest"
    }
  ]
}
```

Sample Response

```
HTTP/1.1 200 OK
Server: Server
Date: Fri, 16 Dec 2016 19:53:56 GMT
Content-Type: application/x-amz-json-1.1
Content-Length: 800
Connection: keep-alive
x-amzn-RequestId: 60dc1ea1-c3c9-11e6-aa04-25c3a5fb1b54

{
  "failures": [],
  "images": [
    {
      "imageId": {
        "imageDigest":
          "sha256:f1d4ae3f7261a72e98c6ebefe9985cf10a0ea5bd762585a43e0700ed99863807",
        "imageTag": "latest"
      },
      "imageManifest": "{\n  \"schemaVersion\": 2,\n  \"mediaType\": \"application/vnd.docker.distribution.manifest.v2+json\",\n  \"config\": {\n    \"mediaType\": \"application/vnd.docker.container.image.v1+json\",\n    \"size\": 1486,\n    \"digest\": \"sha256:5b52b314511a611975c2c65e695d920acdf8ae8848fe0ef00b7d018d1f118b64\"\n  },\n  \"layers\": [\n    {\n      \"mediaType\": \"application/vnd.docker.image.rootfs.diff.tar.gzip\",\n      \"size\": 91768077,\n      \"digest\": \"sha256:8e3fa21c4cc40232e835a6761332d225c7af3235c5755f44ada2ed9d0e4ab7e8\"\n    }\n  ],\n  \"registryId\": \"012345678910\",\n  \"repositoryName\": \"amazonlinux\"\n}"
    ]
  }
}
```

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

Notifies Amazon ECR that the image layer upload has completed for a specified 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 per each new image layer to verify that the upload has completed.

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. 165).

The request accepts the following data in JSON format.

layerDigests (p. 16)

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. 16)

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

Type: String

Pattern: [0-9]{12}

Required: No

repositoryName (p. 16)

The name of the repository to associate with the image layer.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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

Required: Yes

uploadId (p. 16)

The upload ID from a previous [InitiateLayerUpload](#) (p. 80) 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. 17)

The sha256 digest of the image layer.

Type: String

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

registryId (p. 17)

The registry ID associated with the request.

Type: String

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

repositoryName (p. 17)

The repository name associated with the request.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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

uploadId (p. 17)

The upload ID 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 \(p. 167\)](#).

EmptyUploadException

The specified layer upload does not contain any layer parts.

HTTP Status Code: 400

InvalidLayerException

The layer digest calculation performed by Amazon ECR upon receipt of the image layer does not 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

KmsException

The operation failed due to a KMS exception.

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

RepositoryNotFoundException

The specified repository could not be found. Check the spelling of the specified repository and ensure that you are 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

UploadNotFoundException

The upload could not be found, or the specified upload ID is not 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. For more information, see [Amazon ECR repositories](#) in the *Amazon Elastic Container Registry User Guide*.

Request Syntax

```
{
  "encryptionConfiguration": {
    "encryptionType": "string",
    "kmsKey": "string"
  },
  "imageScanningConfiguration": {
    "scanOnPush": boolean
  },
  "imageTagMutability": "string",
  "registryId": "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. 165).

The request accepts the following data in JSON format.

encryptionConfiguration (p. 20)

The encryption configuration for the repository. This determines how the contents of your repository are encrypted at rest.

Type: [EncryptionConfiguration](#) (p. 136) object

Required: No

imageScanningConfiguration (p. 20)

The image scanning configuration for the repository. This determines whether images are scanned for known vulnerabilities after being pushed to the repository.

Type: [ImageScanningConfiguration](#) (p. 148) object

Required: No

imageTagMutability (p. 20)

The tag mutability setting for the repository. If this parameter is omitted, the default setting of `MUTABLE` will be used which will allow image tags to be overwritten. If `IMMUTABLE` is specified, all image tags within the repository will be immutable which will prevent them from being overwritten.

Type: String

Valid Values: `MUTABLE` | `IMMUTABLE`

Required: No

registryId (p. 20)

The AWS account ID associated with the registry to create the repository. If you do not specify a registry, the default registry is assumed.

Type: String

Pattern: [0-9]{12}

Required: No

repositoryName (p. 20)

The name to use for the repository. The repository name may be specified on its own (such as nginx-web-app) or it can be prepended with a namespace to group the repository into a category (such as project-a/nginx-web-app).

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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

Required: Yes

tags (p. 20)

The metadata that you apply to the repository to help you categorize and organize them. Each tag consists of a key and an optional value, both of which you define. 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. 164) objects

Required: No

Response Syntax

```
{
  "repository": {
    "createdAt": number,
    "encryptionConfiguration": {
      "encryptionType": "string",
      "kmsKey": "string"
    },
    "imageScanningConfiguration": {
      "scanOnPush": boolean
    },
    "imageTagMutability": "string",
    "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. 21)

The repository that was created.

Type: [Repository \(p. 161\)](#) object

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 167\)](#).

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

KmsException

The operation failed due to a KMS exception.

HTTP Status Code: 400

LimitExceededException

The operation did not 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

Examples

In the following example or examples, the Authorization header contents (`AUTHPARAMS`) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

This example creates a repository called `sample-repo` in the default registry for an account.

Sample Request

```
POST / HTTP/1.1
Host: api.ecr.us-west-2.amazonaws.com
Accept-Encoding: identity
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.CreateRepository
Content-Type: application/x-amz-json-1.1
User-Agent: aws-cli/1.16.190 Python/3.6.1 Darwin/16.7.0 botocore/1.12.180
X-Amz-Date: 20190715T204735Z
Authorization: AUTHPARAMS
Content-Length: 33

{
  "repositoryName": "sample-repo"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 123a4b56-7c89-01d2-3ef4-example5678f
Content-Type: application/x-amz-json-1.1
Content-Length: 339
Connection: keep-alive

{
  "repository": {
    "repositoryArn": "arn:aws:ecr:us-west-2:012345678910:repository/sample-repo",
    "registryId": "012345678910",
    "repositoryName": "sample-repo",
    "repositoryUri": "012345678910.dkr.ecr.us-west-2.amazonaws.com/sample-repo",
    "createdAt": 1.563223656E9,
    "imageTagMutability": "MUTABLE",
    "imageScanningConfiguration": {
      "scanOnPush": false
    }
  }
}
```

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](#)

DeleteLifecyclePolicy

Deletes the lifecycle policy 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. 165).

The request accepts the following data in JSON format.

registryId (p. 25)

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

Type: String

Pattern: [0-9]{12}

Required: No

repositoryName (p. 25)

The name of the repository.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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

Required: Yes

Response Syntax

```
{  
  "lastEvaluatedAt": number,  
  "lifecyclePolicyText": "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.

lastEvaluatedAt (p. 25)

The time stamp of the last time that the lifecycle policy was run.

Type: Timestamp

lifecyclePolicyText (p. 25)

The JSON lifecycle policy text.

Type: String

Length Constraints: Minimum length of 100. Maximum length of 30720.

registryId (p. 25)

The registry ID associated with the request.

Type: String

Pattern: [0-9]{12}

repositoryName (p. 25)

The repository name associated with the request.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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 \(p. 167\)](#).

InvalidParameterException

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

HTTP Status Code: 400

LifecyclePolicyNotFoundException

The lifecycle policy could not be found, and no policy is set to the repository.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository could not be found. Check the spelling of the specified repository and ensure that you are 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

Examples

In the following example or examples, the Authorization header contents (AUTHPARAMS) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

This example deletes a lifecycle policy for a repository called `project-a/amazon-ecs-sample` in the default registry for an account.

Sample Request

```
POST / HTTP/1.1
Host: ecr.us-west-2.amazonaws.com
Accept-Encoding: identity
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.DeleteLifecyclePolicy
Content-Type: application/x-amz-json-1.1
User-Agent: aws-cli/1.11.144 Python/3.6.1 Darwin/16.6.0 botocore/1.7.2
X-Amz-Date: 20170901T223937Z
Authorization: AUTHPARAMS
Content-Length: 48

{
  "repositoryName": "project-a/amazon-ecs-sample",
}
```

Sample Response

```
HTTP/1.1 200 OK
Server: Server
Date: Fri, 01 Sep 2017 19:42:18 GMT
Content-Type: application/x-amz-json-1.1
Content-Length: 340
Connection: keep-alive
x-amzn-RequestId: 123a4b56-7c89-01d2-3ef4-example5678f

{
  "lastEvaluatedAt": 1.504295007E9,
  "lifecyclePolicyText": "{\n  \"rules\": [\n    {\n      \"rulePriority\": 1,\n      \"description\": \"Expire\nimages older than 14 days\",\n      \"selection\": {\n        \"tagStatus\": \"untagged\",\n        \"countType\":\n        \"sinceImagePushed\",\n        \"countUnit\": \"days\",\n        \"countNumber\": 14\n      },\n      \"action\": {\n        \"type\":\n        \"expire\"\n      }\n    }\n  ],\n  \"registryId\": \"012345678910\",\n  \"repositoryName\": \"project-a/amazon-ecs-sample\"\n}"
```

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](#)

DeleteRegistryPolicy

Deletes the registry permissions policy.

Response Syntax

```
{
  "policyText": "string",
  "registryId": "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. 29)

The contents of the registry permissions policy that was deleted.

Type: String

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

registryId (p. 29)

The registry ID associated with the request.

Type: String

Pattern: [0-9]{12}

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 167\)](#).

InvalidParameterException

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

HTTP Status Code: 400

RegistryPolicyNotFoundException

The registry doesn't have an associated registry policy.

HTTP Status Code: 400

ServerException

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

HTTP Status Code: 500

ValidationException

There was an exception validating this request.

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. If the repository contains images, you must either delete all images in the repository or use the `force` option to delete 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. 165).

The request accepts the following data in JSON format.

force (p. 31)

If a repository contains images, forces the deletion.

Type: Boolean

Required: No

registryId (p. 31)

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

Type: String

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

Required: No

repositoryName (p. 31)

The name of the repository to delete.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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

Required: Yes

Response Syntax

```
{  
  "repository": {  
    "createdAt": number,  
  }  
}
```



```
    "encryptionConfiguration": {  
      "encryptionType": "string",  
      "kmsKey": "string"  
    },  
    "imageScanningConfiguration": {  
      "scanOnPush": boolean  
    },  
    "imageTagMutability": "string",  
    "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. 31)

The repository that was deleted.

Type: [Repository \(p. 161\)](#) object

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 167\)](#).

InvalidParameterException

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

HTTP Status Code: 400

KmsException

The operation failed due to a KMS exception.

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 could not be found. Check the spelling of the specified repository and ensure that you are 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

Examples

In the following example or examples, the Authorization header contents (AUTHPARAMS) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

This example deletes a repository named ubuntu in the default registry for an account.

Sample Request

```
POST / HTTP/1.1
Host: api.ecr.us-west-2.amazonaws.com
Accept-Encoding: identity
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.DeleteRepository
Content-Type: application/x-amz-json-1.1
User-Agent: aws-cli/1.16.190 Python/3.6.1 Darwin/16.7.0 botocore/1.12.180
X-Amz-Date: 20190715T205933Z
Authorization: AUTHPARAMS
Content-Length: 33

{
  "repositoryName": "sample-repo"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 123a4b56-7c89-01d2-3ef4-example5678f
Content-Type: application/x-amz-json-1.1
Content-Length: 252
Connection: keep-alive

{
  "repository": {
    "createdAt": 1563223656E9,
    "registryId": "012345678910",
    "repositoryArn": "arn:aws:ecr:us-west-2:012345678910:repository/sample-repo",
    "repositoryName": "sample-repo",
    "repositoryUri": "012345678910.dkr.ecr.us-west-2.amazonaws.com/sample-repo"
  }
}
```

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 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. 165).

The request accepts the following data in JSON format.

[registryId](#) (p. 35)

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

Type: String

Pattern: [0-9]{12}

Required: No

[repositoryName](#) (p. 35)

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

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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. 35)

The JSON repository policy that was deleted from the repository.

Type: String

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

registryId (p. 35)

The registry ID associated with the request.

Type: String

Pattern: [0-9]{12}

repositoryName (p. 35)

The repository name associated with the request.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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 \(p. 167\)](#).

InvalidParameterException

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

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository could not be found. Check the spelling of the specified repository and ensure that you are performing operations on the correct registry.

HTTP Status Code: 400

RepositoryPolicyNotFoundException

The specified repository and registry combination does not have an associated repository policy.

HTTP Status Code: 400

ServerException

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

HTTP Status Code: 500

Examples

In the following example or examples, the Authorization header contents (AUTHPARAMS) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

This example deletes the repository policy from the ubuntu repository.

Sample Request

```
POST / HTTP/1.1
Host: ecr.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 28
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.DeleteRepositoryPolicy
X-Amz-Date: 20151215T003722Z
User-Agent: aws-cli/1.9.10 Python/2.7.10 Darwin/14.5.0 botocore/1.3.10
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{
  "repositoryName": "ubuntu"
}
```

Sample Response

```
HTTP/1.1 200 OK
Server: Server
Date: Tue, 15 Dec 2015 00:37:22 GMT
Content-Type: application/x-amz-json-1.1
Content-Length: 301
Connection: keep-alive
x-amzn-RequestId: 01817918-a2c4-11e5-a19f-014c7a9aad99

{
  "policyText": "{\n  \"Version\" : \"2012-10-17\",\n  \"Statement\" : [ {\n    \"Sid\" :\n    \"AllowPull\",\n    \"Effect\" : \"Allow\",\n    \"Principal\" : \"*\",\n    \"Action\" :\n    [ \"ecr:BatchGetImage\", \"ecr:GetDownloadUrlForLayer\" ]\n  } ]\n}",
  "registryId": "012345678910",
  "repositoryName": "ubuntu"
}
```

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](#)

DescribeImageReplicationStatus

Returns the replication status for a specified image.

Request Syntax

```
{  
  "imageId": {  
    "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. 165).

The request accepts the following data in JSON format.

imageId (p. 39)

An object with identifying information for an image in an Amazon ECR repository.

Type: [ImageIdentifier](#) (p. 143) object

Required: Yes

registryId (p. 39)

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

Type: String

Pattern: [0-9]{12}

Required: No

repositoryName (p. 39)

The name of the repository that the image is in.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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

Required: Yes

Response Syntax

```
{  
  "imageId": {  

```



```
    "imageDigest": "string",
    "imageTag": "string"
  },
  "replicationStatuses": [
    {
      "failureCode": "string",
      "region": "string",
      "registryId": "string",
      "status": "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.

imageId (p. 39)

An object with identifying information for an image in an Amazon ECR repository.

Type: [ImageIdentifier](#) (p. 143) object

replicationStatuses (p. 39)

The replication status details for the images in the specified repository.

Type: Array of [ImageReplicationStatus](#) (p. 144) objects

repositoryName (p. 39)

The repository name associated with the request.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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. 167).

ImageNotFoundException

The image requested does not 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 could not be found. Check the spelling of the specified repository and ensure that you are 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

ValidationException

There was an exception validating this request.

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](#)

DescribeImages

Returns metadata about the images in a repository.

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 may return a larger image size than the image sizes returned by [DescribeImages](#) (p. 42).

Request Syntax

```
{
  "filter": {
    "tagStatus": "string"
  },
  "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. 165).

The request accepts the following data in JSON format.

[filter](#) (p. 42)

The filter key and value with which to filter your `DescribeImages` results.

Type: [DescribeImagesFilter](#) (p. 135) object

Required: No

[imageIds](#) (p. 42)

The list of image IDs for the requested repository.

Type: Array of [ImageIdentifier](#) (p. 143) objects

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

Required: No

[maxResults](#) (p. 42)

The maximum number of repository results 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. The remaining results of the initial request can be seen by sending another `DescribeImages` request with the returned `nextToken` value. This value can

be between 1 and 1000. If this parameter is not used, then `DescribeImages` returns up to 100 results and a `nextToken` value, if applicable. This option cannot be used when you specify images with `imageIds`.

Type: Integer

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

Required: No

`nextToken` (p. 42)

The `nextToken` value 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. This value is `null` when there are no more results to return. This option cannot be used when you specify images with `imageIds`.

Type: String

Required: No

`registryId` (p. 42)

The AWS account ID associated with the registry that contains the repository in which to describe images. If you do not specify a registry, the default registry is assumed.

Type: String

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

Required: No

`repositoryName` (p. 42)

The repository that contains the images to describe.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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,
      "imageScanFindingsSummary": {
        "findingSeverityCounts": {
          "string" : number
        },
        "imageScanCompletedAt": number,
        "vulnerabilitySourceUpdatedAt": number
      },
    },
  ],
}
```

```
    "imageScanStatus": {
      "description": "string",
      "status": "string"
    },
    "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. 43)

A list of [ImageDetail](#) (p. 140) objects that contain data about the image.

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

nextToken (p. 43)

The nextToken value to include in a future [DescribeImages](#) request. When the results of a [DescribeImages](#) request exceed maxResults, this value can be used to retrieve the next page of results. This value is null when there are no more results to return.

Type: String

Errors

For information about the errors that are common to all actions, see [Common Errors](#) (p. 167).

ImageNotFoundException

The image requested does not 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 could not be found. Check the spelling of the specified repository and ensure that you are 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

Examples

In the following example or examples, the Authorization header contents (AUTHPARAMS) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

This example describes the images in a repository named `ubuntu` in the default account. Note that the image with the digest `sha256:7a64bc9c8843b0a8c8b8a7e4715b7615e4e1b0d8ca3c7e7a76ec8250899c397a` is tagged as `latest` and `xenial`.

Sample Request

```
POST / HTTP/1.1
Host: ecr.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 28
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.DescribeImages
X-Amz-Date: 20161216T193133Z
User-Agent: aws-cli/1.11.22 Python/2.7.12 Darwin/16.3.0 botocore/1.4.79
Content-Type: application/x-amz-json-1.1

Authorization: AUTHPARAMS

{
  "repositoryName": "ubuntu"
}
```

Sample Response

```
HTTP/1.1 200 OK
Server: Server
Date: Fri, 16 Dec 2016 19:31:33 GMT
Content-Type: application/x-amz-json-1.1
Content-Length: 1107
Connection: keep-alive
x-amzn-RequestId: 404826b1-c3c6-11e6-a9e5-e3c203a2f07f

{
  "imageDetails": [
    {
      "imageDigest": "sha256:7c70a5ebcc7fcaa22974a71175ba674efce3951fbec624943c891e9d256927c1",
      "imagePushedAt": 1452721263,
      "imageSizeInBytes": 44194573,
      "registryId": "012345678910",
      "repositoryName": "ubuntu"
    },
    {
```

```
    "imageDigest":  
    "sha256:abdc090336ba4503bd72d0961a4f3d45134900d9a793d3f0c06a64d2555fbab7",  
    "imagePushedAt": 1481916613,  
    "imageSizeInBytes": 39142127,  
    "imageTags": [  
        "precise"  
    ],  
    "registryId": "012345678910",  
    "repositoryName": "ubuntu"  
  },  
  {  
    "imageDigest":  
    "sha256:78dea3347768ba553ee8971cfb2dfa4c3efcd96db84ebfcdda59ba8d6057941b",  
    "imagePushedAt": 1452721207,  
    "imageSizeInBytes": 65747044,  
    "registryId": "012345678910",  
    "repositoryName": "ubuntu"  
  },  
  {  
    "imageDigest":  
    "sha256:7a64bc9c8843b0a8c8b8a7e4715b7615e4e1b0d8ca3c7e7a76ec8250899c397a",  
    "imagePushedAt": 1481916523,  
    "imageSizeInBytes": 50223482,  
    "imageTags": [  
        "xenial",  
        "latest"  
    ],  
    "registryId": "012345678910",  
    "repositoryName": "ubuntu"  
  },  
  {  
    "imageDigest":  
    "sha256:881befbe6f54c1e85029fe3a11554342bf765a0849600ecb8fa2f922798b4925",  
    "imagePushedAt": 1481916647,  
    "imageSizeInBytes": 65770577,  
    "imageTags": [  
        "trusty"  
    ],  
    "registryId": "012345678910",  
    "repositoryName": "ubuntu"  
  }  
]  
}
```

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](#)

DescribeImageScanFindings

Returns the scan findings for the specified image.

Request Syntax

```
{
  "imageId": {
    "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. 165).

The request accepts the following data in JSON format.

imageId (p. 47)

An object with identifying information for an image in an Amazon ECR repository.

Type: [ImageIdentifier](#) (p. 143) object

Required: Yes

maxResults (p. 47)

The maximum number of image scan results returned by `DescribeImageScanFindings` in paginated output. When this parameter is used, `DescribeImageScanFindings` 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 `DescribeImageScanFindings` request with the returned `nextToken` value. This value can be between 1 and 1000. If this parameter is not used, then `DescribeImageScanFindings` 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. 47)

The `nextToken` value returned from a previous paginated `DescribeImageScanFindings` 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. This value is null when there are no more results to return.

Type: String

Required: No

registryId (p. 47)

The AWS account ID associated with the registry that contains the repository in which to describe the image scan findings for. If you do not specify a registry, the default registry is assumed.

Type: String

Pattern: [0-9]{12}

Required: No

repositoryName (p. 47)

The repository for the image for which to describe the scan findings.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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

Required: Yes

Response Syntax

```
{
  "imageId": {
    "imageDigest": "string",
    "imageTag": "string"
  },
  "imageScanFindings": {
    "findings": [
      {
        "attributes": [
          {
            "key": "string",
            "value": "string"
          }
        ],
        "description": "string",
        "name": "string",
        "severity": "string",
        "uri": "string"
      }
    ],
    "findingSeverityCounts": {
      "string" : number
    },
    "imageScanCompletedAt": number,
    "vulnerabilitySourceUpdatedAt": number
  },
  "imageScanStatus": {
    "description": "string",
    "status": "string"
  },
  "nextToken": "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.

imageId (p. 48)

An object with identifying information for an image in an Amazon ECR repository.

Type: [ImageIdentifier](#) (p. 143) object

imageScanFindings (p. 48)

The information contained in the image scan findings.

Type: [ImageScanFindings](#) (p. 146) object

imageScanStatus (p. 48)

The current state of the scan.

Type: [ImageScanStatus](#) (p. 149) object

nextToken (p. 48)

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

Type: String

registryId (p. 48)

The registry ID associated with the request.

Type: String

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

repositoryName (p. 48)

The repository name associated with the request.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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. 167).

ImageNotFoundException

The image requested does not 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 could not be found. Check the spelling of the specified repository and ensure that you are performing operations on the correct registry.

HTTP Status Code: 400

ScanNotFoundException

The specified image scan could not be found. Ensure that image scanning is enabled on the repository and try again.

HTTP Status Code: 400

ServerException

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

HTTP Status Code: 500

Examples

In the following example or examples, the Authorization header contents (AUTHPARAMS) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

This example returns the image scan findings for an image using the image digest in a repository named `sample-repo` in the default registry for an account.

Sample Request

```
POST / HTTP/1.1
Host: ecr.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 141
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.DescribeImageScanFindings
X-Amz-Date: 20161216T201255Z
User-Agent: aws-cli/1.16.310 Python/3.6.1 Darwin/18.7.0 botocore/1.13.46
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{
  "repositoryName": "sample-repo",
  "imageId": {
    "imageDigest":
      "sha256:74b2c688c700ec95a93e478cdb959737c148df3fbf5ea706abe0318726e885e6"
  }
}
```

Sample Response

```
HTTP/1.1 200 OK
```

```
Server: Server
Date: Fri, 24 Jan 2020 03:48:07 GMT
Content-Type: application/x-amz-json-1.1
Content-Length: 33967
Connection: keep-alive
x-amzn-RequestId: 3081a92b-2066-41f8-8a47-0580288ada9e

{
  "imageScanFindings": {
    "findings": [
      {
        "name": "CVE-2019-5188",
        "description": "A code execution vulnerability exists in the directory rehashing functionality of E2fsprogs e2fsck 1.45.4. A specially crafted ext4 directory can cause an out-of-bounds write on the stack, resulting in code execution. An attacker can corrupt a partition to trigger this vulnerability.",
        "uri": "http://people.ubuntu.com/~ubuntu-security/cve/CVE-2019-5188",
        "severity": "MEDIUM",
        "attributes": [
          {
            "key": "package_version",
            "value": "1.44.1-1ubuntu1.1"
          },
          {
            "key": "package_name",
            "value": "e2fsprogs"
          },
          {
            "key": "CVSS2_VECTOR",
            "value": "AV:L/AC:L/Au:N/C:P/I:P/A:P"
          },
          {
            "key": "CVSS2_SCORE",
            "value": "4.6"
          }
        ]
      }
    ],
    "imageScanCompletedAt": 1579839105.0,
    "vulnerabilitySourceUpdatedAt": 1579811117.0,
    "findingSeverityCounts": {
      "MEDIUM": 1
    }
  },
  "registryId": "012345678910",
  "repositoryName": "sample-repo",
  "imageId": {
    "imageDigest":
      "sha256:74b2c688c700ec95a93e478cdb959737c148df3fbf5ea706abe0318726e885e6"
  },
  "imageScanStatus": {
    "status": "COMPLETE",
    "description": "The scan was completed successfully."
  }
}
```

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](#)

DescribeRegistry

Describes the settings for a registry. The replication configuration for a repository can be created or updated with the [PutReplicationConfiguration](#) (p. 109) API action.

Response Syntax

```
{
  "registryId": "string",
  "replicationConfiguration": {
    "rules": [
      {
        "destinations": [
          {
            "region": "string",
            "registryId": "string"
          }
        ],
        "repositoryFilters": [
          {
            "filter": "string",
            "filterType": "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.

registryId (p. 53)

The ID of the registry.

Type: String

Pattern: [0-9]{12}

replicationConfiguration (p. 53)

The replication configuration for the registry.

Type: [ReplicationConfiguration](#) (p. 158) object

Errors

For information about the errors that are common to all actions, see [Common Errors](#) (p. 167).

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

ValidationException

There was an exception validating this request.

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 image repositories in a 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. 165).

The request accepts the following data in JSON format.

maxResults (p. 55)

The maximum number of repository results 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. The remaining results of the initial request can be seen by sending another `DescribeRepositories` request with the returned `nextToken` value. This value can be between 1 and 1000. If this parameter is not used, then `DescribeRepositories` returns up to 100 results and a `nextToken` value, if applicable. This option cannot be used when you specify repositories with `repositoryNames`.

Type: Integer

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

Required: No

nextToken (p. 55)

The `nextToken` value 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. This value is `null` when there are no more results to return. This option cannot be used when you specify repositories with `repositoryNames`.

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. 55)

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

Type: String

Pattern: [0-9]{1,2}

Required: No

repositoryNames (p. 55)

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 256.

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

Required: No

Response Syntax

```
{
  "nextToken": "string",
  "repositories": [
    {
      "createdAt": number,
      "encryptionConfiguration": {
        "encryptionType": "string",
        "kmsKey": "string"
      },
      "imageScanningConfiguration": {
        "scanOnPush": boolean
      },
      "imageTagMutability": "string",
      "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. 56)

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. This value is null when there are no more results to return.

Type: String

repositories (p. 56)

A list of repository objects corresponding to valid repositories.

Type: Array of [Repository](#) (p. 161) objects

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 167\)](#).

InvalidParameterException

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

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository could not be found. Check the spelling of the specified repository and ensure that you are 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

Examples

In the following example or examples, the Authorization header contents (AUTHPARAMS) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

This example describes the repositories in the default registry for an account.

Sample Request

```
POST / HTTP/1.1
Host: api.ecr.us-west-2.amazonaws.com
Accept-Encoding: identity
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.DescribeRepositories
Content-Type: application/x-amz-json-1.1
User-Agent: aws-cli/1.16.190 Python/3.6.1 Darwin/16.7.0 botocore/1.12.180
X-Amz-Date: 20190715T205400Z
Authorization: AUTHPARAMS
Content-Length: 2

{}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 123a4b56-7c89-01d2-3ef4-example5678f
Content-Type: application/x-amz-json-1.1
Content-Length: 1061
```

Connection: keep-alive

```
{
  "repositories":[
    {
      "createdAt":1.563223656E9,
      "imageTagMutability":"MUTABLE",
      "registryId":"012345678910",
      "repositoryArn":"arn:aws:ecr:us-west-2:012345678910:repository/sample-repo",
      "repositoryName":"sample-repo",
      "repositoryUri":"012345678910.dkr.ecr.us-west-2.amazonaws.com/sample-repo"
    },
    {
      "createdAt":1.554824595E9,
      "imageTagMutability":"IMMUTABLE",
      "registryId":"012345678910",
      "repositoryArn":"arn:aws:ecr:us-west-2:012345678910:repository/tagging-test",
      "repositoryName":"tagging-test",
      "repositoryUri":"012345678910.dkr.ecr.us-west-2.amazonaws.com/tagging-test"
    }
  ]
}
```

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 and can be used to access any Amazon ECR registry that your IAM principal has access to. The authorization token is valid for 12 hours.

The `authorizationToken` returned is a base64 encoded string that can be decoded and used in a `docker login` command to authenticate to a registry. The AWS CLI offers an `get-login-password` command that simplifies the login process. For more information, see [Registry authentication](#) in the *Amazon Elastic Container Registry User Guide*.

Request Syntax

```
{
  "registryIds": [ "string" ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#) (p. 165).

The request accepts the following data in JSON format.

`registryIds` (p. 59)

This parameter has been deprecated.

A list of AWS account IDs that are associated with the registries for which to get `AuthorizationData` objects. If you do not specify a registry, the default registry is assumed.

Type: Array of strings

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

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

Required: No

Response Syntax

```
{
  "authorizationData": [
    {
      "authorizationToken": "string",
      "expiresAt": number,
      "proxyEndpoint": "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.

authorizationData (p. 59)

A list of authorization token data objects that correspond to the `registryIds` values in the request.

Type: Array of [AuthorizationData](#) (p. 134) objects

Errors

For information about the errors that are common to all actions, see [Common Errors](#) (p. 167).

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

Examples

In the following example or examples, the Authorization header contents (AUTHPARAMS) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

This example gets an authorization token for your default registry.

Sample Request

```
POST / HTTP/1.1
Host: ecr.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 2
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.GetAuthorizationToken
X-Amz-Date: 20151129T221940Z
User-Agent: aws-cli/1.9.9 Python/2.7.10 Darwin/14.5.0 botocore/1.3.9
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{}
```

Sample Response

```
HTTP/1.1 200 OK
```

```
Server: Server
Date: Sun, 29 Nov 2015 22:19:39 GMT
Content-Type: application/x-amz-json-1.1
Content-Length: 1590
Connection: keep-alive
x-amzn-RequestId: 123a4b56-7c89-01d2-3ef4-example5678f

{
  "authorizationData": [
    {
      "authorizationToken": "QVdT0kNpQzErSHF1ZXZPcUR...",
      "expiresAt": 1448878779.809,
      "proxyEndpoint": "https://012345678910.dkr.ecr.us-east-1.amazonaws.com"
    }
  ]
}
```

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](#)

GetDownloadUrlForLayer

Retrieves the pre-signed Amazon S3 download URL corresponding to an image layer. You can only get URLs for image layers that are referenced in an image.

When an image is pulled, the `GetDownloadUrlForLayer` API is called once per image layer that is not already cached.

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

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

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 165\)](#).

The request accepts the following data in JSON format.

`layerDigest` (p. 62)

The digest of the image layer to download.

Type: String

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

Required: Yes

`registryId` (p. 62)

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

Type: String

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

Required: No

`repositoryName` (p. 62)

The name of the repository that is associated with the image layer to download.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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

Required: Yes

Response Syntax

```
{  
  "downloadUrl": "string",  
  "layerDigest": "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.

downloadUrl (p. 63)

The pre-signed Amazon S3 download URL for the requested layer.

Type: String

layerDigest (p. 63)

The digest of the image layer to download.

Type: String

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

Errors

For information about the errors that are common to all actions, see [Common Errors](#) (p. 167).

InvalidParameterException

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

HTTP Status Code: 400

LayerInaccessibleException

The specified layer is not available because it is not associated with an image. Unassociated image layers may be cleaned up at any time.

HTTP Status Code: 400

LayersNotFoundException

The specified layers could not be found, or the specified layer is not valid for this repository.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository could not be found. Check the spelling of the specified repository and ensure that you are 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](#)

GetLifecyclePolicy

Retrieves the lifecycle 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. 165).

The request accepts the following data in JSON format.

registryId (p. 65)

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

Type: String

Pattern: [0-9]{12}

Required: No

repositoryName (p. 65)

The name of the repository.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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

Required: Yes

Response Syntax

```
{  
  "lastEvaluatedAt": number,  
  "lifecyclePolicyText": "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.

lastEvaluatedAt (p. 65)

The time stamp of the last time that the lifecycle policy was run.

Type: Timestamp

lifecyclePolicyText (p. 65)

The JSON lifecycle policy text.

Type: String

Length Constraints: Minimum length of 100. Maximum length of 30720.

registryId (p. 65)

The registry ID associated with the request.

Type: String

Pattern: [0-9]{12}

repositoryName (p. 65)

The repository name associated with the request.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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 \(p. 167\)](#).

InvalidParameterException

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

HTTP Status Code: 400

LifecyclePolicyNotFoundException

The lifecycle policy could not be found, and no policy is set to the repository.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository could not be found. Check the spelling of the specified repository and ensure that you are 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

Examples

In the following example or examples, the Authorization header contents (AUTHPARAMS) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

This example retrieves the lifecycle policy for a repository called `project-a/amazon-ecs-sample` in the default registry for an account.

Sample Request

```
POST / HTTP/1.1
Host: ecr.us-west-2.amazonaws.com
Accept-Encoding: identity
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.GetLifecyclePolicy
Content-Type: application/x-amz-json-1.1
User-Agent: aws-cli/1.11.144 Python/3.6.1 Darwin/16.6.0 botocore/1.7.2
X-Amz-Date: 20170901T210647Z
Authorization: AUTHPARAMS
Content-Length: 48

{
  "repositoryName": "project-a/amazon-ecs-sample"
}
```

Sample Response

```
HTTP/1.1 200 OK
Server: Server
Date: Fri, 01 Sep 2017 21:06:48 GMT
Content-Type: application/x-amz-json-1.1
Content-Length: 372
Connection: keep-alive
x-amzn-RequestId: 123a4b56-7c89-01d2-3ef4-example5678f

{
  "lastEvaluatedAt": 1.504295007E9,
  "lifecyclePolicyText": "{\n  \"rules\": [\n    {\n      \"rulePriority\": 1,\n      \"description\": \"Expire\nimages older than 14 days\",\n      \"selection\": {\n        \"tagStatus\": \"untagged\",\n        \"countType\":\n        \"sinceImagePushed\",\n        \"countUnit\": \"days\",\n        \"countNumber\": 14\n      },\n      \"action\": {\n        \"type\":\n        \"expire\"\n      }\n    }\n  ],\n  \"registryId\": \"012345678910\",\n  \"repositoryName\": \"project-a/amazon-ecs-sample\"\n}
```

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](#)

GetLifecyclePolicyPreview

Retrieves the results of the lifecycle policy preview request for the specified repository.

Request Syntax

```
{
  "filter": {
    "tagStatus": "string"
  },
  "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. 165).

The request accepts the following data in JSON format.

filter (p. 69)

An optional parameter that filters results based on image tag status and all tags, if tagged.

Type: [LifecyclePolicyPreviewFilter](#) (p. 152) object

Required: No

imageIds (p. 69)

The list of imageIDs to be included.

Type: Array of [ImageIdentifier](#) (p. 143) objects

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

Required: No

maxResults (p. 69)

The maximum number of repository results returned by `GetLifecyclePolicyPreviewRequest` in paginated output. When this parameter is used, `GetLifecyclePolicyPreviewRequest` 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 `GetLifecyclePolicyPreviewRequest` request with the returned `nextToken` value. This value can be between 1 and 1000. If this parameter is not used, then `GetLifecyclePolicyPreviewRequest` returns up to 100 results and a `nextToken` value, if

applicable. This option cannot be used when you specify images with `imageIds`.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 100.

Required: No

`nextToken` (p. 69)

The `nextToken` value returned from a previous paginated `GetLifecyclePolicyPreviewRequest` 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. This value is `null` when there are no more results to return. This option cannot be used when you specify images with `imageIds`.

Type: String

Required: No

`registryId` (p. 69)

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

Type: String

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

Required: No

`repositoryName` (p. 69)

The name of the repository.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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

Required: Yes

Response Syntax

```
{
  "lifecyclePolicyText": "string",
  "nextToken": "string",
  "previewResults": [
    {
      "action": {
        "type": "string"
      },
      "appliedRulePriority": number,
      "imageDigest": "string",
      "imagePushedAt": number,
      "imageTags": [ "string" ]
    }
  ],
  "registryId": "string",
  "repositoryName": "string",
```

```
"status": "string",  
"summary": {  
  "expiringImageTotalCount": 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.

lifecyclePolicyText (p. 70)

The JSON lifecycle policy text.

Type: String

Length Constraints: Minimum length of 100. Maximum length of 30720.

nextToken (p. 70)

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

Type: String

previewResults (p. 70)

The results of the lifecycle policy preview request.

Type: Array of [LifecyclePolicyPreviewResult](#) (p. 153) objects

registryId (p. 70)

The registry ID associated with the request.

Type: String

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

repositoryName (p. 70)

The repository name associated with the request.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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

status (p. 70)

The status of the lifecycle policy preview request.

Type: String

Valid Values: `IN_PROGRESS` | `COMPLETE` | `EXPIRED` | `FAILED`

summary (p. 70)

The list of images that is returned as a result of the action.

Type: [LifecyclePolicyPreviewSummary](#) (p. 155) object

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 167\)](#).

InvalidParameterException

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

HTTP Status Code: 400

LifecyclePolicyPreviewNotFoundException

There is no dry run for this repository.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository could not be found. Check the spelling of the specified repository and ensure that you are 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

Examples

In the following example or examples, the Authorization header contents (AUTHPARAMS) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

This example retrieves the result of a lifecycle policy preview for a repository called `project-a/amazon-ecs-sample` in the default registry for an account.

Sample Request

```
POST / HTTP/1.1
Host: ecr.us-west-2.amazonaws.com
Accept-Encoding: identity
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.GetLifecyclePolicyPreview
Content-Type: application/x-amz-json-1.1
User-Agent: aws-cli/1.11.144 Python/3.6.1 Darwin/16.6.0 botocore/1.7.2
X-Amz-Date: 20170901T222304Z
Authorization: AUTHPARAMS
Content-Length: 48

{
  "repositoryName": "project-a/amazon-ecs-sample"
}
```

```
HTTP/1.1 200 OK
Server: Server
Date: Fri, 01 Sep 2017 22:23:06 GMT
Content-Type: application/x-amz-json-1.1
Content-Length: 640
Connection: keep-alive
x-amzn-RequestId: 123a4b56-7c89-01d2-3ef4-example5678f

{
  "lifecyclePolicyText": "{\n  \"rules\": [\n    {\n      \"rulePriority\n\" : 1,\n      \"description\": \"Expire images older than 14 days\",\n      \"selection\": {\n        \"tagStatus\": \"untagged\",\n        \"countType\": \"sinceImagePushed\",\n        \"countUnit\": \"days\",\n        \"countNumber\": 14\n      },\n      \"action\": {\n        \"type\": \"expire\"\n      }\n    }\n  ]\n}\n",
  "previewResults": [],
  "registryId": "012345678910",
  "repositoryName": "project-a/amazon-ecs-sample",
  "status": "COMPLETE",
  "summary": {"expiringImageTotalCount": 0}
}
```

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

GetRegistryPolicy

Retrieves the permissions policy for a registry.

Response Syntax

```
{  
  "policyText": "string",  
  "registryId": "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. 74)

The JSON text of the permissions policy for a registry.

Type: String

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

registryId (p. 74)

The ID of the registry.

Type: String

Pattern: [0-9]{12}

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 167\)](#).

InvalidParameterException

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

HTTP Status Code: 400

RegistryPolicyNotFoundException

The registry doesn't have an associated registry policy.

HTTP Status Code: 400

ServerException

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

HTTP Status Code: 500

ValidationException

There was an exception validating this request.

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](#)

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. 165).

The request accepts the following data in JSON format.

[registryId](#) (p. 76)

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

Type: String

Pattern: [0-9]{12}

Required: No

[repositoryName](#) (p. 76)

The name of the repository with the policy to retrieve.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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. 76)

The JSON repository policy text associated with the repository.

Type: String

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

registryId (p. 76)

The registry ID associated with the request.

Type: String

Pattern: [0-9]{12}

repositoryName (p. 76)

The repository name associated with the request.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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 \(p. 167\)](#).

InvalidParameterException

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

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository could not be found. Check the spelling of the specified repository and ensure that you are performing operations on the correct registry.

HTTP Status Code: 400

RepositoryPolicyNotFoundException

The specified repository and registry combination does not have an associated repository policy.

HTTP Status Code: 400

ServerException

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

HTTP Status Code: 500

Examples

In the following example or examples, the Authorization header contents (AUTHPARAMS) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

This example gets the repository policy for the ubuntu repository.

Sample Request

```
POST / HTTP/1.1
Host: ecr.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 28
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.GetRepositoryPolicy
X-Amz-Date: 20151215T002404Z
User-Agent: aws-cli/1.9.10 Python/2.7.10 Darwin/14.5.0 botocore/1.3.10
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{
  "repositoryName": "ubuntu"
}
```

Sample Response

```
HTTP/1.1 200 OK
Server: Server
Date: Tue, 15 Dec 2015 00:24:04 GMT
Content-Type: application/x-amz-json-1.1
Content-Length: 301
Connection: keep-alive
x-amzn-RequestId: 25da0b72-a2c2-11e5-8543-ebda6fb1393b

{
  "policyText": "{\n  \"Version\" : \"2012-10-17\",\n  \"Statement\" : [ {\n    \"Sid\" :\n    \"AllowPull\",\n    \"Effect\" : \"Allow\",\n    \"Principal\" : \"*\",\n    \"Action\" :\n    [ \"ecr:BatchGetImage\", \"ecr:GetDownloadUrlForLayer\" ]\n  } ]\n}",
  "registryId": "012345678910",
  "repositoryName": "ubuntu"
}
```

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 per image layer that has not already been uploaded. Whether or not an image layer has been uploaded 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. 165).

The request accepts the following data in JSON format.

registryId (p. 80)

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

Type: String

Pattern: [0-9]{12}

Required: No

repositoryName (p. 80)

The name of the repository to which you intend to upload layers.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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. 80)

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

Type: Long

Valid Range: Minimum value of 0.

uploadId (p. 80)

The upload ID for the layer upload. This parameter is passed to further [UploadLayerPart](#) (p. 128) and [CompleteLayerUpload](#) (p. 16) 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. 167).

InvalidParameterException

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

HTTP Status Code: 400

KmsException

The operation failed due to a KMS exception.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository could not be found. Check the spelling of the specified repository and ensure that you are 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](#)

ListImages

Lists all the image IDs for the specified repository.

You can filter images based on whether or not they are tagged by using the `tagStatus` filter and specifying either `TAGGED`, `UNTAGGED` or `ANY`. For example, you can filter your results to return only `UNTAGGED` images and then pipe that result to a [BatchDeleteImage](#) (p. 7) operation to delete them. Or, you can filter your results to return only `TAGGED` images to list all of the tags in your repository.

Request Syntax

```
{
  "filter": {
    "tagStatus": "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. 165).

The request accepts the following data in JSON format.

filter (p. 83)

The filter key and value with which to filter your `ListImages` results.

Type: [ListImagesFilter](#) (p. 157) object

Required: No

maxResults (p. 83)

The maximum number of image results returned by `ListImages` in paginated output. When this parameter is used, `ListImages` 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 `ListImages` request with the returned `nextToken` value. This value can be between 1 and 1000. If this parameter is not used, then `ListImages` 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. 83)

The `nextToken` value returned from a previous paginated `ListImages` 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. This value is `null` when there are no more results to return.

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. 83)

The AWS account ID associated with the registry that contains the repository in which to list images. If you do not specify a registry, the default registry is assumed.

Type: String

Pattern: [0-9]{12}

Required: No

repositoryName (p. 83)

The repository with image IDs to be listed.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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

Required: Yes

Response Syntax

```
{
  "imageIds": [
    {
      "imageDigest": "string",
      "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.

imageIds (p. 84)

The list of image IDs for the requested repository.

Type: Array of [ImageIdentifier \(p. 143\)](#) objects

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

nextToken (p. 84)

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

Type: String

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 167\)](#).

InvalidParameterException

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

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository could not be found. Check the spelling of the specified repository and ensure that you are 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

Examples

In the following example or examples, the Authorization header contents (AUTHPARAMS) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

This example lists all of the images in the `amazonlinux` repository.

Sample Request

```
POST / HTTP/1.1
Host: ecr.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 33
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.ListImages
X-Amz-Date: 20161216T200542Z
User-Agent: aws-cli/1.11.22 Python/2.7.12 Darwin/16.3.0 botocore/1.4.79
Content-Type: application/x-amz-json-1.1
Authorization: AWUTHPARAMS

{
  "repositoryName": "amazonlinux"
}
```

Sample Response

```
HTTP/1.1 200 OK
```

```
Server: Server
Date: Fri, 16 Dec 2016 20:05:42 GMT
Content-Type: application/x-amz-json-1.1
Content-Length: 235
Connection: keep-alive
x-amzn-RequestId: 05bfc4ac-c3cb-11e6-99fb-b1be070cc24b

{
  "imageIds": [
    {
      "imageDigest":
"sha256:f1d4ae3f7261a72e98c6ebefe9985cf10a0ea5bd762585a43e0700ed99863807",
      "imageTag": "2016.09"
    },
    {
      "imageDigest":
"sha256:f1d4ae3f7261a72e98c6ebefe9985cf10a0ea5bd762585a43e0700ed99863807",
      "imageTag": "latest"
    }
  ]
}
```

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 resource.

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 165\)](#).

The request accepts the following data in JSON format.

resourceArn (p. 87)

The Amazon Resource Name (ARN) that identifies the resource for which to list the tags. Currently, the only supported resource is an Amazon ECR 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. 87)

The tags for the resource.

Type: Array of [Tag \(p. 164\)](#) objects

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 167\)](#).

InvalidParameterException

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

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository could not be found. Check the spelling of the specified repository and ensure that you are 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

Examples

In the following example or examples, the Authorization header contents (AUTHPARAMS) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

This example lists the tags associated with the `sample-repo` repository.

Sample Request

```
POST / HTTP/1.1
Host: ecr.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 81
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.ListTagsForResource
X-Amz-Date: 20161216T201255Z
User-Agent: aws-cli/1.16.310 Python/3.6.1 Darwin/18.7.0 botocore/1.13.46
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{
  "resourceArn": "arn:aws:ecr:us-west-2:012345678910:repository/sample-repo"
}
```

Sample Response

```
HTTP/1.1 200 OK
Server: Server
Date: Fri, 24 Jan 2020 03:48:07 GMT
Content-Type: application/x-amz-json-1.1
Content-Length: 11
Connection: keep-alive
x-amzn-RequestId: 3081a92b-2066-41f8-8a47-0580288ada9e
```

```
{
  "tags": [
    {
      "Key": "environment",
      "Value": "production"
    }
  ]
}
```

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 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 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. 165).

The request accepts the following data in JSON format.

imageDigest (p. 90)

The image digest of the image manifest corresponding to the image.

Type: String

Required: No

imageManifest (p. 90)

The image manifest corresponding to the image to be uploaded.

Type: String

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

Required: Yes

imageManifestMediaType (p. 90)

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

Type: String

Required: No

imageTag (p. 90)

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 \(p. 90\)](#)

The AWS account ID associated with the registry that contains the repository in which to put the image. If you do not specify a registry, the default registry is assumed.

Type: String

Pattern: [0-9]{12}

Required: No

[repositoryName \(p. 90\)](#)

The name of the repository in which to put the image.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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 \(p. 91\)](#)

Details of the image uploaded.

Type: [Image \(p. 138\)](#) object

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 167\)](#).

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 does not 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

KmsException

The operation failed due to a KMS exception.

HTTP Status Code: 400

LayersNotFoundException

The specified layers could not be found, or the specified layer is not valid for this repository.

HTTP Status Code: 400

LimitExceededException

The operation did not 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 does not exist.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository could not be found. Check the spelling of the specified repository and ensure that you are 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

Examples

In the following example or examples, the Authorization header contents (AUTHPARAMS) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

This example puts an image to the amazonlinux repository with the tag 2016.09.

Sample Request

```
POST / HTTP/1.1
Host: ecr.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 653
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.PutImage
X-Amz-Date: 20161216T201255Z
User-Agent: aws-cli/1.11.22 Python/2.7.12 Darwin/16.3.0 botocore/1.4.79
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{
  "imageManifest": "{\n  \"schemaVersion\": 2,\n  \"mediaType\": \"application/\n  vnd.docker.distribution.manifest.v2+json\",\n  \"config\": {\n    \"mediaType\":\n    \"application/vnd.docker.container.image.v1+json\",\n    \"size\": 1486,\n    \"digest\": \"sha256:5b52b314511a611975c2c65e695d920acdf8ae8848fe0ef00b7d018d1f118b64\"\n  },\n  \"layers\": [\n    {\n      \"mediaType\": \"application/\n      vnd.docker.image.rootfs.diff.tar.gzip\",\n      \"size\": 91768077,\n      \"digest\": \"sha256:8e3fa21c4cc40232e835a6761332d225c7af3235c5755f44ada2ed9d0e4ab7e8\"\n    }\n  ]\n}",
  "repositoryName": "amazonlinux",
  "imageTag": "2016.09"
}
```

Sample Response

```
HTTP/1.1 200 OK
Server: Server
Date: Fri, 16 Dec 2016 20:12:56 GMT
Content-Type: application/x-amz-json-1.1
Content-Length: 786
Connection: keep-alive
x-amzn-RequestId: 084038f1-c3cc-11e6-8d10-9da51cf53fd3

{
  "image": {
    "imageId": {
      "imageDigest":
      "sha256:f1d4ae3f7261a72e98c6ebef9985cf10a0ea5bd762585a43e0700ed99863807",
      "imageTag": "2016.09"
    },
    "imageManifest": "{\n  \"schemaVersion\": 2,\n  \"mediaType\": \"application/\n  vnd.docker.distribution.manifest.v2+json\",\n  \"config\": {\n    \"mediaType\":\n    \"application/vnd.docker.container.image.v1+json\",\n    \"size\": 1486,\n    \"digest\": \"sha256:5b52b314511a611975c2c65e695d920acdf8ae8848fe0ef00b7d018d1f118b64\"\n  },\n  \"layers\": [\n    {\n      \"mediaType\": \"application/\n      vnd.docker.image.rootfs.diff.tar.gzip\",\n      \"size\": 91768077,\n      \"digest\": \"sha256:8e3fa21c4cc40232e835a6761332d225c7af3235c5755f44ada2ed9d0e4ab7e8\"\n    }\n  ]\n}"
  }
```

```
\n\"digest\": \"sha256:5b52b314511a611975c2c65e695d920acdf8ae8848fe0ef00b7d018d1f118b64\"\n},\n  \"layers\": [\n    {\n      \"mediaType\": \"application/vnd.docker.image.rootfs.diff.tar.gzip\", \n      \"size\": 91768077, \n      \"digest\": \"sha256:8e3fa21c4cc40232e835a6761332d225c7af3235c5755f44ada2ed9d0e4ab7e8\" \n    }\n  ],\n  \"registryId\": \"012345678910\", \n  \"repositoryName\": \"amazonlinux\"\n}
```

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](#)

PutImageScanningConfiguration

Updates the image scanning configuration for the specified repository.

Request Syntax

```
{
  "imageScanningConfiguration": {
    "scanOnPush": boolean
  },
  "registryId": "string",
  "repositoryName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 165\)](#).

The request accepts the following data in JSON format.

imageScanningConfiguration (p. 95)

The image scanning configuration for the repository. This setting determines whether images are scanned for known vulnerabilities after being pushed to the repository.

Type: [ImageScanningConfiguration \(p. 148\)](#) object

Required: Yes

registryId (p. 95)

The AWS account ID associated with the registry that contains the repository in which to update the image scanning configuration setting. If you do not specify a registry, the default registry is assumed.

Type: String

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

Required: No

repositoryName (p. 95)

The name of the repository in which to update the image scanning configuration setting.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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

Required: Yes

Response Syntax

```
{
```



```
"imageScanningConfiguration": {  
  "scanOnPush": boolean  
},  
"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.

imageScanningConfiguration (p. 95)

The image scanning configuration setting for the repository.

Type: [ImageScanningConfiguration](#) (p. 148) object

registryId (p. 95)

The registry ID associated with the request.

Type: String

Pattern: [0-9]{12}

repositoryName (p. 95)

The repository name associated with the request.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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. 167).

InvalidParameterException

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

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository could not be found. Check the spelling of the specified repository and ensure that you are 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

Examples

In the following example or examples, the Authorization header contents (AUTHPARAMS) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

This example updates the image scanning configuration for the `sample-repo` repository.

Sample Request

```
POST / HTTP/1.1
Host: ecr.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 90
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.PutImageScanningConfiguration
X-Amz-Date: 20161216T201255Z
User-Agent: aws-cli/1.16.310 Python/3.6.1 Darwin/18.7.0 botocore/1.13.46
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{
  "repositoryName": "sample-repo",
  "imageScanningConfiguration": {
    "scanOnPush": true
  }
}
```

Sample Response

```
HTTP/1.1 200 OK
Server: Server
Date: Fri, 24 Jan 2020 03:48:07 GMT
Content-Type: application/x-amz-json-1.1
Content-Length: 114
Connection: keep-alive
x-amzn-RequestId: 3081a92b-2066-41f8-8a47-0580288ada9e

{
  "registryId": "012345678910",
  "repositoryName": "sample-repo",
  "imageScanningConfiguration": {
    "scanOnPush": true
  }
}
```

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](#)

PutImageTagMutability

Updates the image tag mutability settings for the specified repository. For more information, see [Image tag mutability](#) in the *Amazon Elastic Container Registry User Guide*.

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#) (p. 165).

The request accepts the following data in JSON format.

[imageTagMutability](#) (p. 99)

The tag mutability setting for the repository. If `MUTABLE` is specified, image tags can be overwritten. If `IMMUTABLE` is specified, all image tags within the repository will be immutable which will prevent them from being overwritten.

Type: String

Valid Values: `MUTABLE` | `IMMUTABLE`

Required: Yes

[registryId](#) (p. 99)

The AWS account ID associated with the registry that contains the repository in which to update the image tag mutability settings. If you do not specify a registry, the default registry is assumed.

Type: String

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

Required: No

[repositoryName](#) (p. 99)

The name of the repository in which to update the image tag mutability settings.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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

Required: Yes

Response Syntax

```
{
```

```
"imageTagMutability": "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.

imageTagMutability (p. 99)

The image tag mutability setting for the repository.

Type: String

Valid Values: `MUTABLE` | `IMMUTABLE`

registryId (p. 99)

The registry ID associated with the request.

Type: String

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

repositoryName (p. 99)

The repository name associated with the request.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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. 167).

InvalidParameterException

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

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository could not be found. Check the spelling of the specified repository and ensure that you are 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

Examples

In the following example or examples, the Authorization header contents (AUTHPARAMS) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

This example updates the image tag mutability setting for the `sample-repo` repository.

Sample Request

```
POST / HTTP/1.1
Host: ecr.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 73
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.PutImageTagMutability
X-Amz-Date: 20161216T201255Z
User-Agent: aws-cli/1.16.310 Python/3.6.1 Darwin/18.7.0 botocore/1.13.46
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{
  "repositoryName": "sample-repo",
  "imageTagMutability": "IMMUTABLE"
}
```

Sample Response

```
HTTP/1.1 200 OK
Server: Server
Date: Fri, 24 Jan 2020 03:48:07 GMT
Content-Type: application/x-amz-json-1.1
Content-Length: 98
Connection: keep-alive
x-amzn-RequestId: 3081a92b-2066-41f8-8a47-0580288ada9e

{
  "registryId": "012345678910",
  "repositoryName": "sample-repo",
  "imageTagMutability": "IMMUTABLE"
}
```

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](#)

PutLifecyclePolicy

Creates or updates the lifecycle policy for the specified repository. For more information, see [Lifecycle policy template](#).

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#) (p. 165).

The request accepts the following data in JSON format.

[lifecyclePolicyText](#) (p. 103)

The JSON repository policy text to apply to the repository.

Type: String

Length Constraints: Minimum length of 100. Maximum length of 30720.

Required: Yes

[registryId](#) (p. 103)

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

Type: String

Pattern: [0-9]{12}

Required: No

[repositoryName](#) (p. 103)

The name of the repository to receive the policy.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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

Required: Yes

Response Syntax

```
{  
  "lifecyclePolicyText": "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.

lifecyclePolicyText (p. 103)

The JSON repository policy text.

Type: String

Length Constraints: Minimum length of 100. Maximum length of 30720.

registryId (p. 103)

The registry ID associated with the request.

Type: String

Pattern: [0-9]{12}

repositoryName (p. 103)

The repository name associated with the request.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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 \(p. 167\)](#).

InvalidParameterException

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

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository could not be found. Check the spelling of the specified repository and ensure that you are 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

Examples

In the following example or examples, the Authorization header contents (AUTHPARAMS) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

This example creates a lifecycle policy to expire images older than 14 days for a repository called `project-a/amazon-ecs-sample` in the default registry for an account.

Sample Request

```
POST / HTTP/1.1
Host: ecr.us-west-2.amazonaws.com
Accept-Encoding: identity
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.PutLifecyclePolicy
Content-Type: application/x-amz-json-1.1
User-Agent: aws-cli/1.11.144 Python/3.6.1 Darwin/16.6.0 botocore/1.7.2
X-Amz-Date: 20170901T194217Z
Authorization: AUTHPARAMS
Content-Length: 535

{
  "repositoryName": "project-a/amazon-ecs-sample",
  "lifecyclePolicyText": "{\n  \"rules\": [\n    {\n      \"rulePriority\n\": 1,\n      \"description\": \"Expire images older than 14 days\",\n      \"selection\": {\n        \"tagStatus\": \"untagged\",\n        \"countType\": \"sinceImagePushed\",\n        \"countUnit\": \"days\",\n        \"countNumber\": 14\n      },\n      \"action\": {\n        \"type\": \"expire\"\n      }\n    }\n  ]\n}"
```

Sample Response

```
HTTP/1.1 200 OK
Server: Server
Date: Fri, 01 Sep 2017 19:42:18 GMT
Content-Type: application/x-amz-json-1.1
Content-Length: 340
Connection: keep-alive
x-amzn-RequestId: 123a4b56-7c89-01d2-3ef4-example5678f

{
  "lifecyclePolicyText": "{\n  \"rules\": [\n    {\n      \"rulePriority\n\": 1,\n      \"description\": \"Expire\nimages older than 14 days\",\n      \"selection\": {\n        \"tagStatus\": \"untagged\",\n        \"countType\":\n\"sinceImagePushed\",\n        \"countUnit\": \"days\",\n        \"countNumber\": 14\n      },\n      \"action\": {\n        \"type\":\n\"expire\"\n      }\n    }\n  ]\n}",
  "registryId": "012345678910",
  "repositoryName": "project-a/amazon-ecs-sample"
}
```

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](#)

PutRegistryPolicy

Creates or updates the permissions policy for your registry.

A registry policy is used to specify permissions for another AWS account and is used when configuring cross-account replication. For more information, see [Registry permissions](#) in the *Amazon Elastic Container Registry User Guide*.

Request Syntax

```
{  
  "policyText": "string"  
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#) (p. 165).

The request accepts the following data in JSON format.

policyText (p. 107)

The JSON policy text to apply to your registry. The policy text follows the same format as IAM policy text. For more information, see [Registry permissions](#) in the *Amazon Elastic Container Registry User Guide*.

Type: String

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

Required: Yes

Response Syntax

```
{  
  "policyText": "string",  
  "registryId": "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. 107)

The JSON policy text for your registry.

Type: String

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

[registryId \(p. 107\)](#)

The registry ID.

Type: String

Pattern: [0-9]{12}

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 167\)](#).

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

ValidationException

There was an exception validating this request.

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](#)

PutReplicationConfiguration

Creates or updates the replication configuration for a registry. The existing replication configuration for a repository can be retrieved with the [DescribeRegistry](#) (p. 53) API action. The first time the PutReplicationConfiguration API is called, a service-linked IAM role is created in your account for the replication process. For more information, see [Using service-linked roles for Amazon ECR](#) in the *Amazon Elastic Container Registry User Guide*.

Note

When configuring cross-account replication, the destination account must grant the source account permission to replicate. This permission is controlled using a registry permissions policy. For more information, see [PutRegistryPolicy](#) (p. 107).

Request Syntax

```
{
  "replicationConfiguration": {
    "rules": [
      {
        "destinations": [
          {
            "region": "string",
            "registryId": "string"
          }
        ],
        "repositoryFilters": [
          {
            "filter": "string",
            "filterType": "string"
          }
        ]
      }
    ]
  }
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#) (p. 165).

The request accepts the following data in JSON format.

replicationConfiguration (p. 109)

An object representing the replication configuration for a registry.

Type: [ReplicationConfiguration](#) (p. 158) object

Required: Yes

Response Syntax

```
{
  "replicationConfiguration": {
    "rules": [
```

```
{
  "destinations": [
    {
      "region": "string",
      "registryId": "string"
    }
  ],
  "repositoryFilters": [
    {
      "filter": "string",
      "filterType": "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.

replicationConfiguration (p. 109)

The contents of the replication configuration for the registry.

Type: [ReplicationConfiguration](#) (p. 158) object

Errors

For information about the errors that are common to all actions, see [Common Errors](#) (p. 167).

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

ValidationException

There was an exception validating this request.

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](#)

SetRepositoryPolicy

Applies a repository policy to the specified 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. 165).

The request accepts the following data in JSON format.

[force](#) (p. 112)

If the policy you are attempting to set on a repository policy would prevent you from setting another policy in the future, you must force the [SetRepositoryPolicy](#) (p. 112) operation. This is intended to prevent accidental repository lock outs.

Type: Boolean

Required: No

[policyText](#) (p. 112)

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. 112)

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

Type: String

Pattern: [0-9]{12}

Required: No

[repositoryName](#) (p. 112)

The name of the repository to receive the policy.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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. 113)

The JSON repository policy text applied to the repository.

Type: String

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

registryId (p. 113)

The registry ID associated with the request.

Type: String

Pattern: [0-9]{12}

repositoryName (p. 113)

The repository name associated with the request.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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 \(p. 167\)](#).

InvalidParameterException

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

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository could not be found. Check the spelling of the specified repository and ensure that you are 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

Examples

In the following example or examples, the Authorization header contents (AUTHPARAMS) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

The following example sets a repository policy on the ubuntu repository that allows all AWS accounts to pull from it.

Sample Request

```
POST / HTTP/1.1
Host: ecr.us-east-1.amazonaws.com
Accept-Encoding: identity
Content-Length: 223
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.SetRepositoryPolicy
X-Amz-Date: 20151214T235302Z
User-Agent: aws-cli/1.9.10 Python/2.7.10 Darwin/14.5.0 botocore/1.3.10
Content-Type: application/x-amz-json-1.1
Authorization: AWUTHPARAMS

{
  "policyText": "{\n  \"Version\": \"2012-10-17\",\n  \"Statement\": [\n    {\n      \"Sid\": \"AllowPull\",\n      \"Effect\": \"Allow\",\n      \"Action\": [\n        \"ecr:BatchGetImage\",\n        \"ecr:GetDownloadUrlForLayer\"\n      ],\n      \"Principal\": \"*\n    }\n  ]\n}",
  "repositoryName": "ubuntu"
}
```

Sample Response

```
HTTP/1.1 200 OK
Server: Server
Date: Mon, 14 Dec 2015 23:53:02 GMT
Content-Type: application/x-amz-json-1.1
Content-Length: 301
Connection: keep-alive
x-amzn-RequestId: cfc3ead9-a2bd-11e5-91c7-7126cb670c2b

{
  "policyText": "{\n  \"Version\" : \"2012-10-17\",\n  \"Statement\" : [\n    {\n      \"Sid\" : \"AllowPull\",\n      \"Effect\" : \"Allow\",\n      \"Principal\" : \"*\n    }\n  ]\n  \"Action\" : [\n    \"ecr:BatchGetImage\", \"ecr:GetDownloadUrlForLayer\"\n  ]\n}",
  "registryId": "012345678910",
  "repositoryName": "ubuntu"
}
```

```
}
```

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](#)

StartImageScan

Starts an image vulnerability scan. An image scan can only be started once per 24 hours on an individual image. This limit includes if an image was scanned on initial push. For more information, see [Image scanning](#) in the *Amazon Elastic Container Registry User Guide*.

Request Syntax

```
{
  "imageId": {
    "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. 165).

The request accepts the following data in JSON format.

imageId (p. 116)

An object with identifying information for an image in an Amazon ECR repository.

Type: [ImageIdentifier](#) (p. 143) object

Required: Yes

registryId (p. 116)

The AWS account ID associated with the registry that contains the repository in which to start an image scan request. If you do not specify a registry, the default registry is assumed.

Type: String

Pattern: [0-9]{12}

Required: No

repositoryName (p. 116)

The name of the repository that contains the images to scan.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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

Required: Yes

Response Syntax

```
{
```

```
"imageId": {  
  "imageDigest": "string",  
  "imageTag": "string"  
},  
"imageScanStatus": {  
  "description": "string",  
  "status": "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.

imageId (p. 116)

An object with identifying information for an image in an Amazon ECR repository.

Type: [ImageIdentifier](#) (p. 143) object

imageScanStatus (p. 116)

The current state of the scan.

Type: [ImageScanStatus](#) (p. 149) object

registryId (p. 116)

The registry ID associated with the request.

Type: String

Pattern: [0-9]{12}

repositoryName (p. 116)

The repository name associated with the request.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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. 167).

ImageNotFoundException

The image requested does not 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

LimitExceededException

The operation did not 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

RepositoryNotFoundException

The specified repository could not be found. Check the spelling of the specified repository and ensure that you are 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

UnsupportedImageTypeException

The image is of a type that cannot be scanned.

HTTP Status Code: 400

Examples

In the following example or examples, the Authorization header contents (AUTHPARAMS) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

This example starts an image scan for and specified by the image digest in the `sample-repo` repository.

Sample Request

```
POST / HTTP/1.1
Host: ecr.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 141
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.StartImageScan
X-Amz-Date: 20161216T201255Z
User-Agent: aws-cli/1.16.310 Python/3.6.1 Darwin/18.7.0 botocore/1.13.46
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{
  "repositoryName": "sample-repo",
  "imageId": {
    "imageDigest":
      "sha256:74b2c688c700ec95a93e478cdb959737c148df3fbf5ea706abe0318726e885e6"
  }
}
```

```
}
```

Sample Response

```
HTTP/1.1 200 OK
Server: Server
Date: Fri, 24 Jan 2020 03:48:07 GMT
Content-Type: application/x-amz-json-1.1
Content-Length: 208
Connection: keep-alive
x-amzn-RequestId: 3081a92b-2066-41f8-8a47-0580288ada9e

{
  "registryId": "012345678910",
  "repositoryName": "sample-repo",
  "imageId": {
    "imageDigest":
      "sha256:74b2c688c700ec95a93e478cdb959737c148df3fbf5ea706abe0318726e885e6"
  },
  "imageScanStatus": {
    "status": "IN_PROGRESS"
  }
}
```

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](#)

StartLifecyclePolicyPreview

Starts a preview of a lifecycle policy for the specified repository. This allows you to see the results before associating the lifecycle policy with the repository.

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#) (p. 165).

The request accepts the following data in JSON format.

lifecyclePolicyText (p. 120)

The policy to be evaluated against. If you do not specify a policy, the current policy for the repository is used.

Type: String

Length Constraints: Minimum length of 100. Maximum length of 30720.

Required: No

registryId (p. 120)

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

Type: String

Pattern: [0-9]{12}

Required: No

repositoryName (p. 120)

The name of the repository to be evaluated.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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

Required: Yes

Response Syntax

```
{
```

```
"lifecyclePolicyText": "string",  
"registryId": "string",  
"repositoryName": "string",  
"status": "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.

`lifecyclePolicyText` (p. 120)

The JSON repository policy text.

Type: String

Length Constraints: Minimum length of 100. Maximum length of 30720.

`registryId` (p. 120)

The registry ID associated with the request.

Type: String

Pattern: [0-9]{12}

`repositoryName` (p. 120)

The repository name associated with the request.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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

`status` (p. 120)

The status of the lifecycle policy preview request.

Type: String

Valid Values: IN_PROGRESS | COMPLETE | EXPIRED | FAILED

Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 167\)](#).

`InvalidParameterException`

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

HTTP Status Code: 400

`LifecyclePolicyNotFoundException`

The lifecycle policy could not be found, and no policy is set to the repository.

HTTP Status Code: 400

LifecyclePolicyPreviewInProgressException

The previous lifecycle policy preview request has not completed. Wait and try again.

HTTP Status Code: 400

RepositoryNotFoundException

The specified repository could not be found. Check the spelling of the specified repository and ensure that you are 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

Examples

In the following example or examples, the Authorization header contents (AUTHPARAMS) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see [Signature Version 4 Signing Process](#) in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the [AWS Command Line Interface \(AWS CLI\)](#) or one of the [AWS SDKs](#) to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Example

This example creates a lifecycle policy preview to expire images older than 14 days for a repository called `project-a/amazon-ecs-sample` in the default registry for an account.

Sample Request

```
POST / HTTP/1.1
Host: ecr.us-west-2.amazonaws.com
Accept-Encoding: identity
X-Amz-Target: AmazonEC2ContainerRegistry_V20150921.StartLifecyclePolicyPreview
Content-Type: application/x-amz-json-1.1
User-Agent: aws-cli/1.11.144 Python/3.6.1 Darwin/16.6.0 botocore/1.7.2
X-Amz-Date: 20170901T221604Z
Authorization: AUTHPARAMS
Content-Length: 535

{
  "repositoryName": "project-a/amazon-ecs-sample",
  "lifecyclePolicyText": "{\n  \"rules\": [\n    {\n      \"rulePriority\n\": 1,\n      \"description\": \"Expire images older than 14 days\",\n      \"selection\": {\n        \"tagStatus\": \"untagged\",\n        \"countType\": \"sinceImagePushed\",\n        \"countUnit\": \"days\",\n        \"countNumber\": 14\n      },\n      \"action\": {\n        \"type\": \"expire\"\n      }\n    }\n  ]\n}"
```

Sample Response

```
HTTP/1.1 200 OK
```

```
Server: Server
Date: Fri, 01 Sep 2017 22:16:05 GMT
Content-Type: application/x-amz-json-1.1
Content-Length: 583
Connection: keep-alive
x-amzn-RequestId: 123a4b56-7c89-01d2-3ef4-example5678f

{
    "lifecyclePolicyText": "{\n    \"rules\": [\n        {\n            \"rulePriority\n\": 1,\n            \"description\": \"Expire images older than 14 days\",\n            \"selection\": {\n                \"tagStatus\": \"untagged\",\n                \"countType\": \"sinceImagePushed\",\n                \"countUnit\": \"days\",\n                \"countNumber\": 14\n            },\n            \"action\": {\n                \"type\": \"expire\"\n            }\n        }\n    ]\n}\n",
    "registryId": "012345678910",
    "repositoryName": "project-a/amazon-ecs-sample",
    "status": "IN_PROGRESS"
}
```

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

Adds specified tags to a resource with the specified ARN. Existing tags on a resource are not changed if they are not specified in the request parameters.

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. 165).

The request accepts the following data in JSON format.

resourceArn (p. 124)

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

Type: String

Required: Yes

tags (p. 124)

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. 164) objects

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. 167).

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 could not be found. Check the spelling of the specified repository and ensure that you are 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 \(p. 165\)](#).

The request accepts the following data in JSON format.

resourceArn (p. 126)

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

Type: String

Required: Yes

tagKeys (p. 126)

The keys of the tags to be removed.

Type: Array of strings

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 \(p. 167\)](#).

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 could not be found. Check the spelling of the specified repository and ensure that you are 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 (or about 20MB). The UploadLayerPart API is called once per 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. 165).

The request accepts the following data in JSON format.

layerPartBlob (p. 128)

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. 128)

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

Type: Long

Valid Range: Minimum value of 0.

Required: Yes

partLastByte (p. 128)

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. 128)

The AWS account ID associated with the registry to which you are uploading layer parts. If you do not specify a registry, the default registry is assumed.

Type: String

Pattern: [0-9]{12}

Required: No

repositoryName (p. 128)

The name of the repository to which you are uploading layer parts.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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

Required: Yes

uploadId (p. 128)

The upload ID from a previous [InitiateLayerUpload](#) (p. 80) 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. 129)

The integer value of the last byte received in the request.

Type: Long

Valid Range: Minimum value of 0.

registryId (p. 129)

The registry ID associated with the request.

Type: String

Pattern: [0-9]{12}

repositoryName (p. 129)

The repository name associated with the request.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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

uploadId (p. 129)

The upload ID 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 \(p. 167\)](#).

InvalidLayerPartException

The layer part size is not valid, or the first byte specified is not 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

KmsException

The operation failed due to a KMS exception.

HTTP Status Code: 400

LimitExceededException

The operation did not 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

RepositoryNotFoundException

The specified repository could not be found. Check the spelling of the specified repository and ensure that you are 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

UploadNotFoundException

The upload could not be found, or the specified upload ID is not 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 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:

- [Attribute](#) (p. 133)
- [AuthorizationData](#) (p. 134)
- [DescribeImagesFilter](#) (p. 135)
- [EncryptionConfiguration](#) (p. 136)
- [Image](#) (p. 138)
- [ImageDetail](#) (p. 140)
- [ImageFailure](#) (p. 142)
- [ImageIdentifier](#) (p. 143)
- [ImageReplicationStatus](#) (p. 144)
- [ImageScanFinding](#) (p. 145)
- [ImageScanFindings](#) (p. 146)
- [ImageScanFindingsSummary](#) (p. 147)
- [ImageScanningConfiguration](#) (p. 148)
- [ImageScanStatus](#) (p. 149)
- [Layer](#) (p. 150)
- [LayerFailure](#) (p. 151)
- [LifecyclePolicyPreviewFilter](#) (p. 152)
- [LifecyclePolicyPreviewResult](#) (p. 153)
- [LifecyclePolicyPreviewSummary](#) (p. 155)
- [LifecyclePolicyRuleAction](#) (p. 156)
- [ListImagesFilter](#) (p. 157)
- [ReplicationConfiguration](#) (p. 158)
- [ReplicationDestination](#) (p. 159)
- [ReplicationRule](#) (p. 160)
- [Repository](#) (p. 161)
- [RepositoryFilter](#) (p. 163)
- [Tag](#) (p. 164)

Attribute

This data type is used in the [ImageScanFinding](#) (p. 145) data type.

Contents

key

The attribute key.

Type: String

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

Required: Yes

value

The value assigned to the attribute key.

Type: String

Length Constraints: Minimum length of 1. 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](#)

AuthorizationData

An object representing authorization data for an Amazon ECR registry.

Contents

authorizationToken

A base64-encoded string that contains authorization data for the specified Amazon ECR registry. When the string is decoded, it is presented in the format `user:password` for private 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

proxyEndpoint

The registry URL to use for this authorization token in a `docker login` command. The Amazon ECR registry URL format is `https://aws_account_id.dkr.ecr.region.amazonaws.com`. For example, `https://012345678910.dkr.ecr.us-east-1.amazonaws.com..`

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](#)

DescribeImagesFilter

An object representing a filter on a [DescribeImages](#) (p. 42) operation.

Contents

tagStatus

The tag status with which to filter your [DescribeImages](#) (p. 42) results. You can filter results based on whether they are `TAGGED` or `UNTAGGED`.

Type: String

Valid Values: `TAGGED` | `UNTAGGED` | `ANY`

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](#)

EncryptionConfiguration

The encryption configuration for the repository. This determines how the contents of your repository are encrypted at rest.

By default, when no encryption configuration is set or the `AES256` encryption type is used, Amazon ECR uses server-side encryption with Amazon S3-managed encryption keys which encrypts your data at rest using an AES-256 encryption algorithm. This does not require any action on your part.

For more control over the encryption of the contents of your repository, you can use server-side encryption with AWS Key Management Service key stored in AWS Key Management Service (AWS KMS) to encrypt your images. For more information, see [Amazon ECR encryption at rest](#) in the *Amazon Elastic Container Registry User Guide*.

Contents

encryptionType

The encryption type to use.

If you use the `KMS` encryption type, the contents of the repository will be encrypted using server-side encryption with AWS Key Management Service key stored in AWS KMS. When you use AWS KMS to encrypt your data, you can either use the default AWS managed AWS KMS key for Amazon ECR, or specify your own AWS KMS key, which you already created. For more information, see [Protecting data using server-side encryption with an AWS KMS key stored in AWS Key Management Service \(SSE-KMS\)](#) in the *Amazon Simple Storage Service Console Developer Guide*.

If you use the `AES256` encryption type, Amazon ECR uses server-side encryption with Amazon S3-managed encryption keys which encrypts the images in the repository using an AES-256 encryption algorithm. For more information, see [Protecting data using server-side encryption with Amazon S3-managed encryption keys \(SSE-S3\)](#) in the *Amazon Simple Storage Service Console Developer Guide*.

Type: String

Valid Values: `AES256` | `KMS`

Required: Yes

kmsKey

If you use the `KMS` encryption type, specify the AWS KMS key to use for encryption. The alias, key ID, or full ARN of the AWS KMS key can be specified. The key must exist in the same Region as the repository. If no key is specified, the default AWS managed AWS KMS key for Amazon ECR will be used.

Type: String

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

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 representing an Amazon ECR image.

Contents

imageId

An object containing the image tag and image digest associated with an image.

Type: [ImageIdentifier](#) (p. 143) object

Required: No

imageManifest

The image manifest 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 associated with the registry containing the image.

Type: String

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

Required: No

repositoryName

The name of the repository associated with the image.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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 returned by a [DescribeImages](#) (p. 42) 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, at which the current image was pushed to the repository.

Type: Timestamp

Required: No

imageScanFindingsSummary

A summary of the last completed image scan.

Type: [ImageScanFindingsSummary](#) (p. 147) object

Required: No

imageScanStatus

The current state of the scan.

Type: [ImageScanStatus](#) (p. 149) object

Required: No

imageSizeInBytes

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

If the image is a manifest list, this will be 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 may return a larger image size than the image sizes returned by [DescribeImages](#) (p. 42).

Type: Long

Required: No

imageTags

The list of tags 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 associated with the registry to which this image belongs.

Type: String

Pattern: [0-9]{12}

Required: No

repositoryName

The name of the repository to which this image belongs.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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 representing an Amazon ECR image failure.

Contents

failureCode

The code 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 associated with the failure.

Type: [ImageIdentifier](#) (p. 143) 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 image in an Amazon ECR repository.

Contents

imageDigest

The sha256 digest of the image manifest.

Type: String

Required: No

imageTag

The tag 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](#)

ImageReplicationStatus

The status of the replication process for an image.

Contents

failureCode

The failure code for a replication that has failed.

Type: String

Required: No

region

The destination Region for the image replication.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 25.

Pattern: `[0-9a-z-]{2,25}`

Required: No

registryId

The AWS account ID associated with the registry to which the image belongs.

Type: String

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

Required: No

status

The image replication status.

Type: String

Valid Values: `IN_PROGRESS` | `COMPLETE` | `FAILED`

Required: No

See Also

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

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

ImageScanFinding

Contains information about an image scan finding.

Contents

attributes

A collection of attributes of the host from which the finding is generated.

Type: Array of [Attribute \(p. 133\)](#) objects

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Required: No

description

The description of the finding.

Type: String

Required: No

name

The name associated with the finding, usually a CVE number.

Type: String

Required: No

severity

The finding severity.

Type: String

Valid Values: `INFORMATIONAL` | `LOW` | `MEDIUM` | `HIGH` | `CRITICAL` | `UNDEFINED`

Required: No

uri

A link containing additional details about the security vulnerability.

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](#)

ImageScanFindings

The details of an image scan.

Contents

findings

The findings from the image scan.

Type: Array of [ImageScanFinding](#) (p. 145) objects

Required: No

findingSeverityCounts

The image vulnerability counts, sorted by severity.

Type: String to integer map

Valid Keys: `INFORMATIONAL` | `LOW` | `MEDIUM` | `HIGH` | `CRITICAL` | `UNDEFINED`

Valid Range: Minimum value of 0.

Required: No

imageScanCompletedAt

The time of the last completed image scan.

Type: Timestamp

Required: No

vulnerabilitySourceUpdatedAt

The time when the vulnerability data was last scanned.

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](#)

ImageScanFindingsSummary

A summary of the last completed image scan.

Contents

findingSeverityCounts

The image vulnerability counts, sorted by severity.

Type: String to integer map

Valid Keys: `INFORMATIONAL` | `LOW` | `MEDIUM` | `HIGH` | `CRITICAL` | `UNDEFINED`

Valid Range: Minimum value of 0.

Required: No

imageScanCompletedAt

The time of the last completed image scan.

Type: Timestamp

Required: No

vulnerabilitySourceUpdatedAt

The time when the vulnerability data was last scanned.

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](#)

ImageScanningConfiguration

The image scanning configuration for a repository.

Contents

scanOnPush

The setting that determines whether images are scanned after being pushed to a repository. If set to `true`, images will be scanned after being pushed. If this parameter is not specified, it will default to `false` and images will not be scanned unless a scan is manually started with the [API_StartImageScan](#) API.

Type: Boolean

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](#)

ImageScanStatus

The current status of an image scan.

Contents

description

The description of the image scan status.

Type: String

Required: No

status

The current state of an image scan.

Type: String

Valid Values: `IN_PROGRESS` | `COMPLETE` | `FAILED`

Required: No

See Also

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

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

Layer

An object representing 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 representing an Amazon ECR image layer failure.

Contents

failureCode

The failure code 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 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](#)

LifecyclePolicyPreviewFilter

The filter for the lifecycle policy preview.

Contents

tagStatus

The tag status of the image.

Type: String

Valid Values: `TAGGED` | `UNTAGGED` | `ANY`

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](#)

LifecyclePolicyPreviewResult

The result of the lifecycle policy preview.

Contents

action

The type of action to be taken.

Type: [LifecyclePolicyRuleAction](#) (p. 156) object

Required: No

appliedRulePriority

The priority of the applied rule.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

imageDigest

The sha256 digest of the image manifest.

Type: String

Required: No

imagePushedAt

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

Type: Timestamp

Required: No

imageTags

The list of tags associated with this image.

Type: Array of strings

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](#)

LifecyclePolicyPreviewSummary

The summary of the lifecycle policy preview request.

Contents

expiringImageTotalCount

The number of expiring images.

Type: Integer

Valid Range: Minimum value of 0.

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](#)

LifecyclePolicyRuleAction

The type of action to be taken.

Contents

type

The type of action to be taken.

Type: String

Valid Values: `EXPIRE`

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](#)

ListImagesFilter

An object representing a filter on a [ListImages](#) (p. 83) operation.

Contents

tagStatus

The tag status with which to filter your [ListImages](#) (p. 83) results. You can filter results based on whether they are `TAGGED` or `UNTAGGED`.

Type: String

Valid Values: `TAGGED` | `UNTAGGED` | `ANY`

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](#)

ReplicationConfiguration

The replication configuration for a registry.

Contents

rules

An array of objects representing the replication destinations and repository filters for a replication configuration.

Type: Array of [ReplicationRule](#) (p. 160) objects

Array Members: Minimum number of 0 items. Maximum number of 10 items.

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](#)

ReplicationDestination

An array of objects representing the destination for a replication rule.

Contents

region

The Region to replicate to.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 25.

Pattern: `[0-9a-z-]{2,25}`

Required: Yes

registryId

The AWS account ID of the Amazon ECR private registry to replicate to. When configuring cross-Region replication within your own registry, specify your own account ID.

Type: String

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

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](#)

ReplicationRule

An array of objects representing the replication destinations and repository filters for a replication configuration.

Contents

destinations

An array of objects representing the destination for a replication rule.

Type: Array of [ReplicationDestination](#) (p. 159) objects

Array Members: Minimum number of 0 items. Maximum number of 25 items.

Required: Yes

repositoryFilters

An array of objects representing the filters for a replication rule. Specifying a repository filter for a replication rule provides a method for controlling which repositories in a private registry are replicated.

Type: Array of [RepositoryFilter](#) (p. 163) objects

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

Required: No

See Also

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

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

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

encryptionConfiguration

The encryption configuration for the repository. This determines how the contents of your repository are encrypted at rest.

Type: [EncryptionConfiguration](#) (p. 136) object

Required: No

imageScanningConfiguration

The image scanning configuration for a repository.

Type: [ImageScanningConfiguration](#) (p. 148) object

Required: No

imageTagMutability

The tag mutability setting for the repository.

Type: String

Valid Values: `MUTABLE` | `IMMUTABLE`

Required: No

registryId

The AWS account ID associated with the 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 256.

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](#)

RepositoryFilter

The filter settings used with image replication. Specifying a repository filter to a replication rule provides a method for controlling which repositories in a private registry are replicated. If no repository filter is specified, all images in the repository are replicated.

Contents

filter

The repository filter details. When the `PREFIX_MATCH` filter type is specified, this value is required and should be the repository name prefix to configure replication for.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 256.

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

Required: Yes

filterType

The repository filter type. The only supported value is `PREFIX_MATCH`, which is a repository name prefix specified with the `filter` parameter.

Type: String

Valid Values: `PREFIX_MATCH`

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](#)

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, both of which you define. 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

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

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'THHMMSS'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