



API Reference

# AWS Elemental MediaStore



**API Version 2017-09-01**

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

# AWS Elemental MediaStore: API Reference

Copyright © 2024 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

<b>Welcome</b> .....	<b>1</b>
AWS Elemental MediaStore .....	1
AWS Elemental MediaStore Data Plane .....	1
<b>Actions</b> .....	<b>2</b>
AWS Elemental MediaStore .....	3
CreateContainer .....	4
DeleteContainer .....	8
DeleteContainerPolicy .....	11
DeleteCorsPolicy .....	13
DeleteLifecyclePolicy .....	15
DeleteMetricPolicy .....	17
DescribeContainer .....	19
GetContainerPolicy .....	22
GetCorsPolicy .....	25
GetLifecyclePolicy .....	28
GetMetricPolicy .....	31
ListContainers .....	34
ListTagsForResource .....	37
PutContainerPolicy .....	40
PutCorsPolicy .....	43
PutLifecyclePolicy .....	46
PutMetricPolicy .....	49
StartAccessLogging .....	52
StopAccessLogging .....	54
TagResource .....	56
UntagResource .....	59
AWS Elemental MediaStore Data Plane .....	61
DeleteObject .....	62
DescribeObject .....	65
GetObject .....	69
ListItems .....	74
PutObject .....	78
<b>Data Types</b> .....	<b>83</b>
AWS Elemental MediaStore .....	83

---

Container .....	84
CorsRule .....	87
MetricPolicy .....	90
MetricPolicyRule .....	92
Tag .....	94
AWS Elemental MediaStore Data Plane .....	95
Item .....	96
<b>Common Parameters .....</b>	<b>98</b>
<b>Common Errors .....</b>	<b>101</b>

# Welcome

## AWS Elemental MediaStore

An AWS Elemental MediaStore container is a namespace that holds folders and objects. You use a container endpoint to create, read, and delete objects.

## AWS Elemental MediaStore Data Plane

An AWS Elemental MediaStore asset is an object, similar to an object in the Amazon S3 service. Objects are the fundamental entities that are stored in AWS Elemental MediaStore.

# Actions

The following actions are supported by AWS Elemental MediaStore:

- [CreateContainer](#)
- [DeleteContainer](#)
- [DeleteContainerPolicy](#)
- [DeleteCorsPolicy](#)
- [DeleteLifecyclePolicy](#)
- [DeleteMetricPolicy](#)
- [DescribeContainer](#)
- [GetContainerPolicy](#)
- [GetCorsPolicy](#)
- [GetLifecyclePolicy](#)
- [GetMetricPolicy](#)
- [ListContainers](#)
- [ListTagsForResource](#)
- [PutContainerPolicy](#)
- [PutCorsPolicy](#)
- [PutLifecyclePolicy](#)
- [PutMetricPolicy](#)
- [StartAccessLogging](#)
- [StopAccessLogging](#)
- [TagResource](#)
- [UntagResource](#)

The following actions are supported by AWS Elemental MediaStore Data Plane:

- [DeleteObject](#)
- [DescribeObject](#)
- [GetObject](#)

- [ListItems](#)
- [PutObject](#)

## AWS Elemental MediaStore

The following actions are supported by AWS Elemental MediaStore:

- [CreateContainer](#)
- [DeleteContainer](#)
- [DeleteContainerPolicy](#)
- [DeleteCorsPolicy](#)
- [DeleteLifecyclePolicy](#)
- [DeleteMetricPolicy](#)
- [DescribeContainer](#)
- [GetContainerPolicy](#)
- [GetCorsPolicy](#)
- [GetLifecyclePolicy](#)
- [GetMetricPolicy](#)
- [ListContainers](#)
- [ListTagsForResource](#)
- [PutContainerPolicy](#)
- [PutCorsPolicy](#)
- [PutLifecyclePolicy](#)
- [PutMetricPolicy](#)
- [StartAccessLogging](#)
- [StopAccessLogging](#)
- [TagResource](#)
- [UntagResource](#)

# CreateContainer

Service: AWS Elemental MediaStore

Creates a storage container to hold objects. A container is similar to a bucket in the Amazon S3 service.

## Request Syntax

```
{
  "ContainerName": "string",
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

## Request Parameters

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

The request accepts the following data in JSON format.

### ContainerName

The name for the container. The name must be from 1 to 255 characters. Container names must be unique to your AWS account within a specific region. As an example, you could create a container named `movies` in every region, as long as you don't have an existing container with that name.

Type: String

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

Pattern: `[\w- ]+`

Required: Yes

### Tags

An array of key:value pairs that you define. These values can be anything that you want. Typically, the tag key represents a category (such as "environment") and the tag value



represents a specific value within that category (such as "test," "development," or "production"). You can add up to 50 tags to each container. For more information about tagging, including naming and usage conventions, see [Tagging Resources in MediaStore](#).

Type: Array of [Tag](#) objects

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

Required: No

## Response Syntax

```
{
  "Container": {
    "AccessLoggingEnabled": boolean,
    "ARN": "string",
    "CreationTime": number,
    "Endpoint": "string",
    "Name": "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.

### [Container](#)

**ContainerARN:** The Amazon Resource Name (ARN) of the newly created container. The ARN has the following format: arn:aws:<region>:<account that owns this container>:container/<name of container>. For example: arn:aws:mediastore:us-west-2:111122223333:container/movies

**ContainerName:** The container name as specified in the request.

**CreationTime:** Unix time stamp.

**Status:** The status of container creation or deletion. The status is one of the following: CREATING, ACTIVE, or DELETING. While the service is creating the container, the status is CREATING. When an endpoint is available, the status changes to ACTIVE.

The return value does not include the container's endpoint. To make downstream requests, you must obtain this value by using [DescribeContainer](#) or [ListContainers](#).

Type: [Container](#) object

## Errors

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

### ContainerInUseException

The container that you specified in the request already exists or is being updated.

HTTP Status Code: 400

### InternalServerError

The service is temporarily unavailable.

HTTP Status Code: 500

### LimitExceededException

A service limit has been exceeded.

HTTP Status Code: 400

## Examples

### Example Request

The following example shows how to create a container with the name `movies`:

```
POST / HTTP/1.1
Host: mediastore.us-west-2.amazonaws.com
x-amz-Date: 20170323T120000Z
x-amz-target:ElementalMediaStoreService_20160711.CreateContainer
content-type:application/x-amz-json-1.1
Content-Length:30
Authorization: AWS4-HMAC-SHA256
  Credential=AKIAIOSFODNN7EXAMPLE/20141123/us-west-2/mediastore/aws4_request,
  SignedHeaders=host;x-amz-date;x-amz-mediastore-version,
```

```
Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
{
  "ContainerName": "movies",
}
```

## Example Response

This example illustrates one usage of CreateContainer.

```
HTTP/1.1
{
  "Container":
  {
    "ContainerARN": "arn:aws:mediastore:us-west-2:111122223333:container/movies"
    "ContainerName": "movies"
    "CreationTime": "1491522511.486",
    "Status": "CREATING"
  }
}
```

## 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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# DeleteContainer

Service: AWS Elemental MediaStore

Deletes the specified container. Before you make a `DeleteContainer` request, delete any objects in the container or in any folders in the container. You can delete only empty containers.

## Request Syntax

```
{
  "ContainerName": "string"
}
```

## Request Parameters

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

The request accepts the following data in JSON format.

### ContainerName

The name of the container to delete.

Type: String

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

Pattern: `[\w- ]+`

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](#).

### **ContainerInUseException**

The container that you specified in the request already exists or is being updated.

HTTP Status Code: 400

### **ContainerNotFoundException**

The container that you specified in the request does not exist.

HTTP Status Code: 400

### **InternalServerError**

The service is temporarily unavailable.

HTTP Status Code: 500

## **Examples**

### **Example Request**

The following command has no return value:

```
POST / HTTP/1.1
content-type:application/x-amz-json-1.1
host:mediastore.us-west-2.amazonaws.com
x-amz-date:20170620T230417Z
x-amz-target:ElementalMediaStoreService_20160711.DeleteContainer

{
  "ContainerName":"movies"
}
```

## **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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)

- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# DeleteContainerPolicy

Service: AWS Elemental MediaStore

Deletes the access policy that is associated with the specified container.

## Request Syntax

```
{  
  "ContainerName": "string"  
}
```

## Request Parameters

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

The request accepts the following data in JSON format.

### ContainerName

The name of the container that holds the policy.

Type: String

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

Pattern: `[\w- ]+`

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](#).

### ContainerInUseException

The container that you specified in the request already exists or is being updated.

HTTP Status Code: 400

## ContainerNotFoundException

The container that you specified in the request does not exist.

HTTP Status Code: 400

## InternalServerError

The service is temporarily unavailable.

HTTP Status Code: 500

## PolicyNotFoundException

The policy that you specified in the request does not exist.

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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)



# DeleteCorsPolicy

Service: AWS Elemental MediaStore

Deletes the cross-origin resource sharing (CORS) configuration information that is set for the container.

To use this operation, you must have permission to perform the `MediaStore:DeleteCorsPolicy` action. The container owner has this permission by default and can grant this permission to others.

## Request Syntax

```
{  
  "ContainerName": "string"  
}
```

## Request Parameters

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

The request accepts the following data in JSON format.

### ContainerName

The name of the container to remove the policy from.

Type: String

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

Pattern: `[\w- ]+`

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](#).

## ContainerInUseException

The container that you specified in the request already exists or is being updated.

HTTP Status Code: 400

## ContainerNotFoundException

The container that you specified in the request does not exist.

HTTP Status Code: 400

## CorsPolicyNotFoundException

The CORS policy that you specified in the request does not exist.

HTTP Status Code: 400

## InternalServerError

The service is temporarily unavailable.

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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# DeleteLifecyclePolicy

Service: AWS Elemental MediaStore

Removes an object lifecycle policy from a container. It takes up to 20 minutes for the change to take effect.

## Request Syntax

```
{
  "ContainerName": "string"
}
```

## Request Parameters

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

The request accepts the following data in JSON format.

### ContainerName

The name of the container that holds the object lifecycle policy.

Type: String

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

Pattern: `[\w- ]+`

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](#).

### ContainerInUseException

The container that you specified in the request already exists or is being updated.

HTTP Status Code: 400

### **ContainerNotFoundException**

The container that you specified in the request does not exist.

HTTP Status Code: 400

### **InternalServerError**

The service is temporarily unavailable.

HTTP Status Code: 500

### **PolicyNotFoundException**

The policy that you specified in the request does not exist.

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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# DeleteMetricPolicy

Service: AWS Elemental MediaStore

Deletes the metric policy that is associated with the specified container. If there is no metric policy associated with the container, MediaStore doesn't send metrics to CloudWatch.

## Request Syntax

```
{
  "ContainerName": "string"
}
```

## Request Parameters

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

The request accepts the following data in JSON format.

### ContainerName

The name of the container that is associated with the metric policy that you want to delete.

Type: String

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

Pattern: `[\w- ]+`

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](#).

### ContainerInUseException

The container that you specified in the request already exists or is being updated.

HTTP Status Code: 400

### **ContainerNotFoundException**

The container that you specified in the request does not exist.

HTTP Status Code: 400

### **InternalServerError**

The service is temporarily unavailable.

HTTP Status Code: 500

### **PolicyNotFoundException**

The policy that you specified in the request does not exist.

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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# DescribeContainer

Service: AWS Elemental MediaStore

Retrieves the properties of the requested container. This request is commonly used to retrieve the endpoint of a container. An endpoint is a value assigned by the service when a new container is created. A container's endpoint does not change after it has been assigned. The `DescribeContainer` request returns a single `Container` object based on `ContainerName`. To return all `Container` objects that are associated with a specified AWS account, use [ListContainers](#).

## Request Syntax

```
{  
  "ContainerName": "string"  
}
```

## Request Parameters

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

The request accepts the following data in JSON format.

### ContainerName

The name of the container to query.

Type: String

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

Pattern: `[\w- ]+`

Required: No

## Response Syntax

```
{  
  "Container": {  
    "AccessLoggingEnabled": boolean,  
    "ARN": "string",  
    "CreationTime": number,  
  }
```

```
"Endpoint": "string",  
"Name": "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.

### Container

The name of the queried container.

Type: [Container](#) object

## Errors

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

### **ContainerNotFoundException**

The container that you specified in the request does not exist.

HTTP Status Code: 400

### **InternalServerError**

The service is temporarily unavailable.

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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# GetContainerPolicy

Service: AWS Elemental MediaStore

Retrieves the access policy for the specified container. For information about the data that is included in an access policy, see the [AWS Identity and Access Management User Guide](#).

## Request Syntax

```
{
  "ContainerName": "string"
}
```

## Request Parameters

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

The request accepts the following data in JSON format.

### ContainerName

The name of the container.

Type: String

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

Pattern: `[\w- ]+`

Required: Yes

## Response Syntax

```
{
  "Policy": "string"
}
```

## Response Elements

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

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

## **Policy**

The contents of the access policy.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 8192.

Pattern: `[\x00-\x7F]+`

## **Errors**

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

### **ContainerInUseException**

The container that you specified in the request already exists or is being updated.

HTTP Status Code: 400

### **ContainerNotFoundException**

The container that you specified in the request does not exist.

HTTP Status Code: 400

### **InternalServerError**

The service is temporarily unavailable.

HTTP Status Code: 500

### **PolicyNotFoundException**

The policy that you specified in the request does not exist.

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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# GetCorsPolicy

Service: AWS Elemental MediaStore

Returns the cross-origin resource sharing (CORS) configuration information that is set for the container.

To use this operation, you must have permission to perform the `MediaStore:GetCorsPolicy` action. By default, the container owner has this permission and can grant it to others.

## Request Syntax

```
{
  "ContainerName": "string"
}
```

## Request Parameters

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

The request accepts the following data in JSON format.

### ContainerName

The name of the container that the policy is assigned to.

Type: String

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

Pattern: `[\w- ]+`

Required: Yes

## Response Syntax

```
{
  "CorsPolicy": [
    {
      "AllowedHeaders": [ "string" ],
      "AllowedMethods": [ "string" ],
      "AllowedOrigins": [ "string" ],
    }
  ]
}
```

```
    "ExposeHeaders": [ "string" ],  
    "MaxAgeSeconds": 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.

### CorsPolicy

The CORS policy assigned to the container.

Type: Array of [CorsRule](#) 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](#).

### **ContainerInUseException**

The container that you specified in the request already exists or is being updated.

HTTP Status Code: 400

### **ContainerNotFoundException**

The container that you specified in the request does not exist.

HTTP Status Code: 400

### **CorsPolicyNotFoundException**

The CORS policy that you specified in the request does not exist.

HTTP Status Code: 400

### **InternalServerError**

The service is temporarily unavailable.

## 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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# GetLifecyclePolicy

Service: AWS Elemental MediaStore

Retrieves the object lifecycle policy that is assigned to a container.

## Request Syntax

```
{  
  "ContainerName": "string"  
}
```

## Request Parameters

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

The request accepts the following data in JSON format.

### ContainerName

The name of the container that the object lifecycle policy is assigned to.

Type: String

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

Pattern: `[\w- ]+`

Required: Yes

## Response Syntax

```
{  
  "LifecyclePolicy": "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.



## **LifecyclePolicy**

The object lifecycle policy that is assigned to the container.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 8192.

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

## **Errors**

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

### **ContainerInUseException**

The container that you specified in the request already exists or is being updated.

HTTP Status Code: 400

### **ContainerNotFoundException**

The container that you specified in the request does not exist.

HTTP Status Code: 400

### **InternalServerError**

The service is temporarily unavailable.

HTTP Status Code: 500

### **PolicyNotFoundException**

The policy that you specified in the request does not exist.

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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# GetMetricPolicy

Service: AWS Elemental MediaStore

Returns the metric policy for the specified container.

## Request Syntax

```
{
  "ContainerName": "string"
}
```

## Request Parameters

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

The request accepts the following data in JSON format.

### ContainerName

The name of the container that is associated with the metric policy.

Type: String

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

Pattern: `[\w-]+`

Required: Yes

## Response Syntax

```
{
  "MetricPolicy": {
    "ContainerLevelMetrics": "string",
    "MetricPolicyRules": [
      {
        "ObjectGroup": "string",
        "ObjectGroupName": "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.

### MetricPolicy

The metric policy that is associated with the specific container.

Type: [MetricPolicy](#) object

## Errors

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

### **ContainerInUseException**

The container that you specified in the request already exists or is being updated.

HTTP Status Code: 400

### **ContainerNotFoundException**

The container that you specified in the request does not exist.

HTTP Status Code: 400

### **InternalServerError**

The service is temporarily unavailable.

HTTP Status Code: 500

### **PolicyNotFoundException**

The policy that you specified in the request does not exist.

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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# ListContainers

Service: AWS Elemental MediaStore

Lists the properties of all containers in AWS Elemental MediaStore.

You can query to receive all the containers in one response. Or you can include the `MaxResults` parameter to receive a limited number of containers in each response. In this case, the response includes a token. To get the next set of containers, send the command again, this time with the `NextToken` parameter (with the returned token as its value). The next set of responses appears, with a token if there are still more containers to receive.

See also [DescribeContainer](#), which gets the properties of one container.

## Request Syntax

```
{
  "MaxResults": number,
  "NextToken": "string"
}
```

## Request Parameters

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

The request accepts the following data in JSON format.

### [MaxResults](#)

Enter the maximum number of containers in the response. Use from 1 to 255 characters.

Type: Integer

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

Required: No

### [NextToken](#)

Only if you used `MaxResults` in the first command, enter the token (which was included in the previous response) to obtain the next set of containers. This token is included in a response only if there actually are more containers to list.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: [0-9A-Za-z=/+]+

Required: No

## Response Syntax

```
{
  "Containers": [
    {
      "AccessLoggingEnabled": boolean,
      "ARN": "string",
      "CreationTime": number,
      "Endpoint": "string",
      "Name": "string",
      "Status": "string"
    }
  ],
  "NextToken": "string"
}
```

## Response Elements

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

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

### Containers

The names of the containers.

Type: Array of [Container](#) objects

### NextToken

NextToken is the token to use in the next call to `ListContainers`. This token is returned only if you included the `MaxResults` tag in the original command, and only if there are still containers to return.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `[0-9A-Za-z=/+]+`

## Errors

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

### InternalServerError

The service is temporarily unavailable.

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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)



# ListTagsForResource

Service: AWS Elemental MediaStore

Returns a list of the tags assigned to the specified container.

## Request Syntax

```
{
  "Resource": "string"
}
```

## Request Parameters

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

The request accepts the following data in JSON format.

### Resource

The Amazon Resource Name (ARN) for the container.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `arn:aws:mediastore:[a-z]+-[a-z]+-\d:\d{12}:container/[\w-]{1,255}`

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

An array of key:value pairs that are assigned to the container.

Type: Array of [Tag](#) objects

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

## Errors

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

### **ContainerInUseException**

The container that you specified in the request already exists or is being updated.

HTTP Status Code: 400

### **ContainerNotFoundException**

The container that you specified in the request does not exist.

HTTP Status Code: 400

### **InternalServerError**

The service is temporarily unavailable.

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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# PutContainerPolicy

Service: AWS Elemental MediaStore

Creates an access policy for the specified container to restrict the users and clients that can access it. For information about the data that is included in an access policy, see the [AWS Identity and Access Management User Guide](#).

For this release of the REST API, you can create only one policy for a container. If you enter PutContainerPolicy twice, the second command modifies the existing policy.

## Request Syntax

```
{
  "ContainerName": "string",
  "Policy": "string"
}
```

## Request Parameters

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

The request accepts the following data in JSON format.

### ContainerName

The name of the container.

Type: String

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

Pattern: `[\w- ]+`

Required: Yes

### Policy

The contents of the policy, which includes the following:

- One Version tag
- One Statement tag that contains the standard tags for the policy.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 8192.

Pattern: `[\x00-\x7F]+`

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](#).

### ContainerInUseException

The container that you specified in the request already exists or is being updated.

HTTP Status Code: 400

### ContainerNotFoundException

The container that you specified in the request does not exist.

HTTP Status Code: 400

### InternalServerError

The service is temporarily unavailable.

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 v2](#)

- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# PutCorsPolicy

Service: AWS Elemental MediaStore

Sets the cross-origin resource sharing (CORS) configuration on a container so that the container can service cross-origin requests. For example, you might want to enable a request whose origin is `http://www.example.com` to access your AWS Elemental MediaStore container at `my.example.container.com` by using the browser's XMLHttpRequest capability.

To enable CORS on a container, you attach a CORS policy to the container. In the CORS policy, you configure rules that identify origins and the HTTP methods that can be executed on your container. The policy can contain up to 398,000 characters. You can add up to 100 rules to a CORS policy. If more than one rule applies, the service uses the first applicable rule listed.

To learn more about CORS, see [Cross-Origin Resource Sharing \(CORS\) in AWS Elemental MediaStore](#).

## Request Syntax

```
{
  "ContainerName": "string",
  "CorsPolicy": [
    {
      "AllowedHeaders": [ "string" ],
      "AllowedMethods": [ "string" ],
      "AllowedOrigins": [ "string" ],
      "ExposeHeaders": [ "string" ],
      "MaxAgeSeconds": number
    }
  ]
}
```

## Request Parameters

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

The request accepts the following data in JSON format.

### ContainerName

The name of the container that you want to assign the CORS policy to.

Type: String

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

Pattern: `[\w- ]+`

Required: Yes

### **CorsPolicy**

The CORS policy to apply to the container.

Type: Array of [CorsRule](#) objects

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

Required: Yes

## **Response Elements**

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

## **Errors**

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

### **ContainerInUseException**

The container that you specified in the request already exists or is being updated.

HTTP Status Code: 400

### **ContainerNotFoundException**

The container that you specified in the request does not exist.

HTTP Status Code: 400

### **InternalServerError**

The service is temporarily unavailable.

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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# PutLifecyclePolicy

Service: AWS Elemental MediaStore

Writes an object lifecycle policy to a container. If the container already has an object lifecycle policy, the service replaces the existing policy with the new policy. It takes up to 20 minutes for the change to take effect.

For information about how to construct an object lifecycle policy, see [Components of an Object Lifecycle Policy](#).

## Request Syntax

```
{
  "ContainerName": "string",
  "LifecyclePolicy": "string"
}
```

## Request Parameters

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

The request accepts the following data in JSON format.

### [ContainerName](#)

The name of the container that you want to assign the object lifecycle policy to.

Type: String

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

Pattern: `[\w- ]+`

Required: Yes

### [LifecyclePolicy](#)

The object lifecycle policy to apply to the container.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 8192.

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

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](#).

### ContainerInUseException

The container that you specified in the request already exists or is being updated.

HTTP Status Code: 400

### ContainerNotFoundException

The container that you specified in the request does not exist.

HTTP Status Code: 400

### InternalServerError

The service is temporarily unavailable.

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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)

- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# PutMetricPolicy

Service: AWS Elemental MediaStore

The metric policy that you want to add to the container. A metric policy allows AWS Elemental MediaStore to send metrics to Amazon CloudWatch. It takes up to 20 minutes for the new policy to take effect.

## Request Syntax

```
{
  "ContainerName": "string",
  "MetricPolicy": {
    "ContainerLevelMetrics": "string",
    "MetricPolicyRules": [
      {
        "ObjectGroup": "string",
        "ObjectGroupName": "string"
      }
    ]
  }
}
```

## Request Parameters

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

The request accepts the following data in JSON format.

### ContainerName

The name of the container that you want to add the metric policy to.

Type: String

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

Pattern: `[\w- ]+`

Required: Yes

### MetricPolicy

The metric policy that you want to associate with the container. In the policy, you must indicate whether you want MediaStore to send container-level metrics. You can also include up to five

rules to define groups of objects that you want MediaStore to send object-level metrics for. If you include rules in the policy, construct each rule with both of the following:

- An object group that defines which objects to include in the group. The definition can be a path or a file name, but it can't have more than 900 characters. Valid characters are: a-z, A-Z, 0-9, \_ (underscore), = (equal), : (colon), . (period), - (hyphen), ~ (tilde), / (forward slash), and \* (asterisk). Wildcards (\*) are acceptable.
- An object group name that allows you to refer to the object group. The name can't have more than 30 characters. Valid characters are: a-z, A-Z, 0-9, and \_ (underscore).

Type: [MetricPolicy](#) object

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](#).

### ContainerInUseException

The container that you specified in the request already exists or is being updated.

HTTP Status Code: 400

### ContainerNotFoundException

The container that you specified in the request does not exist.

HTTP Status Code: 400

### InternalServerError

The service is temporarily unavailable.

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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# StartAccessLogging

Service: AWS Elemental MediaStore

Starts access logging on the specified container. When you enable access logging on a container, MediaStore delivers access logs for objects stored in that container to Amazon CloudWatch Logs.

## Request Syntax

```
{
  "ContainerName": "string"
}
```

## Request Parameters

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

The request accepts the following data in JSON format.

### ContainerName

The name of the container that you want to start access logging on.

Type: String

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

Pattern: `[\w- ]+`

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](#).

### ContainerInUseException

The container that you specified in the request already exists or is being updated.



HTTP Status Code: 400

### **ContainerNotFoundException**

The container that you specified in the request does not exist.

HTTP Status Code: 400

### **InternalServerError**

The service is temporarily unavailable.

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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# StopAccessLogging

Service: AWS Elemental MediaStore

Stops access logging on the specified container. When you stop access logging on a container, MediaStore stops sending access logs to Amazon CloudWatch Logs. These access logs are not saved and are not retrievable.

## Request Syntax

```
{  
  "ContainerName": "string"  
}
```

## Request Parameters

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

The request accepts the following data in JSON format.

### ContainerName

The name of the container that you want to stop access logging on.

Type: String

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

Pattern: `[\w- ]+`

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](#).

### ContainerInUseException

The container that you specified in the request already exists or is being updated.

HTTP Status Code: 400

### **ContainerNotFoundException**

The container that you specified in the request does not exist.

HTTP Status Code: 400

### **InternalServerError**

The service is temporarily unavailable.

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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# TagResource

Service: AWS Elemental MediaStore

Adds tags to the specified AWS Elemental MediaStore container. Tags are key:value pairs that you can associate with AWS resources. For example, the tag key might be "customer" and the tag value might be "companyA." You can specify one or more tags to add to each container. You can add up to 50 tags to each container. For more information about tagging, including naming and usage conventions, see [Tagging Resources in MediaStore](#).

## Request Syntax

```
{
  "Resource": "string",
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

## Request Parameters

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

The request accepts the following data in JSON format.

### Resource

The Amazon Resource Name (ARN) for the container.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `arn:aws:mediastore:[a-z]+-[a-z]+-\d:\d{12}:container/[\w-]{1,255}`

Required: Yes

### Tags

An array of key:value pairs that you want to add to the container. You need to specify only the tags that you want to add or update. For example, suppose a container already has two tags

(customer:CompanyA and priority:High). You want to change the priority tag and also add a third tag (type:Contract). For TagResource, you specify the following tags: priority:Medium, type:Contract. The result is that your container has three tags: customer:CompanyA, priority:Medium, and type:Contract.

Type: Array of [Tag](#) objects

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

Required: Yes

## Response Elements

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

## Errors

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

### ContainerInUseException

The container that you specified in the request already exists or is being updated.

HTTP Status Code: 400

### ContainerNotFoundException

The container that you specified in the request does not exist.

HTTP Status Code: 400

### InternalServerError

The service is temporarily unavailable.

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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# UntagResource

Service: AWS Elemental MediaStore

Removes tags from the specified container. You can specify one or more tags to remove.

## Request Syntax

```
{
  "Resource": "string",
  "TagKeys": [ "string" ]
}
```

## Request Parameters

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

The request accepts the following data in JSON format.

### Resource

The Amazon Resource Name (ARN) for the container.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `arn:aws:mediastore:[a-z]+-[a-z]+-\d:\d{12}:container/[\w-]{1,255}`

Required: Yes

### TagKeys

A comma-separated list of keys for tags that you want to remove from the container. For example, if your container has two tags (customer:CompanyA and priority:High) and you want to remove one of the tags (priority:High), you specify the key for the tag that you want to remove (priority).

Type: Array of strings

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

Pattern: `[\p{L}\p{Z}\p{N}_.:/=+\-@]*`

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](#).

### ContainerInUseException

The container that you specified in the request already exists or is being updated.

HTTP Status Code: 400

### ContainerNotFoundException

The container that you specified in the request does not exist.

HTTP Status Code: 400

### InternalServerError

The service is temporarily unavailable.

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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)



- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

## AWS Elemental MediaStore Data Plane

The following actions are supported by AWS Elemental MediaStore Data Plane:

- [DeleteObject](#)
- [DescribeObject](#)
- [GetObject](#)
- [ListItems](#)
- [PutObject](#)

# DeleteObject

Service: AWS Elemental MediaStore Data Plane

Deletes an object at the specified path.

## Request Syntax

```
DELETE /Path+ HTTP/1.1
```

## URI Request Parameters

The request uses the following URI parameters.

### Path

The path (including the file name) where the object is stored in the container. Format: <folder name>/<folder name>/<file name>

Length Constraints: Minimum length of 1. Maximum length of 900.

Pattern: (?:[A-Za-z0-9\_=:\.\\-\~]+/){0,10}[A-Za-z0-9\_=:\.\\-\~]+

Required: Yes

## Request Body

The request does not have a request body.

## Response Syntax

```
HTTP/1.1 200
```

## Response Elements

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

## Errors

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

## ContainerNotFoundException

The specified container was not found for the specified account.

HTTP Status Code: 404

## InternalServerError

The service is temporarily unavailable.

HTTP Status Code: 500

## ObjectNotFoundException

Could not perform an operation on an object that does not exist.

HTTP Status Code: 404

## Examples

### Example

The following request deletes the file `m1aw.avi` that is in the folder `premium/canada`:

```
DELETE premium/canada/m1aw.avi
Host: aaabbbcccdddee.files.mediastore-us-west-2.com
x-amz-Date: 20170323T120000Z
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/us-
west-2/mediastore/aws4_request,SignedHeaders=host;x-amz-date;x-amz-mediastore-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
Content-Length: 0
x-amz-mediastore-version: 2016-07-11
```

## 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 v2](#)

- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# DescribeObject

Service: AWS Elemental MediaStore Data Plane

Gets the headers for an object at the specified path.

## Request Syntax

```
HEAD /Path+ HTTP/1.1
```

## URI Request Parameters

The request uses the following URI parameters.

### Path

The path (including the file name) where the object is stored in the container. Format: <folder name>/<folder name>/<file name>

Length Constraints: Minimum length of 1. Maximum length of 900.

Pattern: (?:[A-Za-z0-9\_=:\.\\-\~]+/){0,10}[A-Za-z0-9\_=:\.\\-\~]+

Required: Yes

## Request Body

The request does not have a request body.

## Response Syntax

```
HTTP/1.1 200  
ETag: ETag  
Content-Type: ContentType  
Content-Length: ContentLength  
Cache-Control: CacheControl  
Last-Modified: LastModified
```

## Response Elements

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

The response returns the following HTTP headers.

### CacheControl

An optional `CacheControl` header that allows the caller to control the object's cache behavior. Headers can be passed in as specified in the HTTP at <https://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.9>.

Headers with a custom user-defined value are also accepted.

### ContentLength

The length of the object in bytes.

Valid Range: Minimum value of 0.

### ContentType

The content type of the object.

Pattern: `^[\w\-\.\+]{1,255}$`

### ETag

The ETag that represents a unique instance of the object.

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

Pattern: `[0-9A-Fa-f]+`

### LastModified

The date and time that the object was last modified.

## **Errors**

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

### **ContainerNotFoundException**

The specified container was not found for the specified account.

HTTP Status Code: 404

### **InternalServerError**

The service is temporarily unavailable.

HTTP Status Code: 500

## ObjectNotFoundException

Could not perform an operation on an object that does not exist.

HTTP Status Code: 404

## Examples

### Example

The following request gets the headers for the file `mLaw.avi` from the folder `premium/canada` in the container that is identified by the endpoint that is specified in the `Host` header.

```
HEAD premium/canada/mLaw.avi
Host: aaabbbcccddee.files.mediastore-us-west-2.com
x-amz-Date: 20170323T120000Z
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/us-
west-2/mediastore/aws4_request,SignedHeaders=host;x-amz-date;x-amz-mediastore-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
Content-Length: 0
x-amz-mediastore-version: 2016-07-11
```

## 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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)





# GetObject

Service: AWS Elemental MediaStore Data Plane

Downloads the object at the specified path. If the object's upload availability is set to streaming, AWS Elemental MediaStore downloads the object even if it's still uploading the object.

When an end user or a content delivery network (CDN) like Amazon CloudFront retrieves an object from MediaStore, the service returns HTTP headers that affect the caching behavior of the object. To learn more about these headers, see [MediaStore's interaction with HTTP caches](#) in the AWS Elemental MediaStore User Guide.

## Request Syntax

```
GET /Path+ HTTP/1.1  
Range: Range
```

## URI Request Parameters

The request uses the following URI parameters.

### Path

The path (including the file name) where the object is stored in the container. Format: <folder name>/<folder name>/<file name>

For example, to upload the file `m1aw.avi` to the folder path `premium\canada` in the container `movies`, enter the path `premium/canada/m1aw.avi`.

Do not include the container name in this path.

If the path includes any folders that don't exist yet, the service creates them. For example, suppose you have an existing `premium/usa` subfolder. If you specify `premium/canada`, the service creates a `canada` subfolder in the `premium` folder. You then have two subfolders, `usa` and `canada`, in the `premium` folder.

There is no correlation between the path to the source and the path (folders) in the container in AWS Elemental MediaStore.

For more information about folders and how they exist in a container, see the [AWS Elemental MediaStore User Guide](#).

The file name is the name that is assigned to the file that you upload. The file can have the same name inside and outside of AWS Elemental MediaStore, or it can have the same name. The file name can include or omit an extension.

Length Constraints: Minimum length of 1. Maximum length of 900.

Pattern: `(?:[A-Za-z0-9_=:.\-~]+/){0,10}[A-Za-z0-9_=:.\-~]+`

Required: Yes

## Range

The range bytes of an object to retrieve. For more information about the Range header, see <http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.35>. AWS Elemental MediaStore ignores this header for partially uploaded objects that have streaming upload availability.

Pattern: `^bytes=(?:\d+\-\d*|\d*\-\d+)$`

## Request Body

The request does not have a request body.

## Response Syntax

```
HTTP/1.1 Status Code
Cache-Control: CacheControl
Content-Range: ContentRange
Content-Length: ContentLength
Content-Type: ContentType
ETag: ETag
Last-Modified: LastModified
```

*Body*

## Response Elements

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

### Status Code

The HTML status code of the request. Status codes ranging from 200 to 299 indicate success. All other status codes indicate the type of error that occurred.

The response returns the following HTTP headers.

### CacheControl

An optional `CacheControl` header that allows the caller to control the object's cache behavior. Headers can be passed in as specified in the HTTP spec at <https://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.9>.

Headers with a custom user-defined value are also accepted.

### ContentLength

The length of the object in bytes.

Valid Range: Minimum value of 0.

### ContentRange

The range of bytes to retrieve.

Pattern: `^bytes=\d+\-\d+/\d+$`

### ContentType

The content type of the object.

Pattern: `^[\\w\\-\\\\.\\+]{1,255}$`

### ETag

The ETag that represents a unique instance of the object.

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

Pattern: `[0-9A-Fa-f]+`

### LastModified

The date and time that the object was last modified.

The response returns the following as the HTTP body.

### Body

The bytes of the object.

## Errors

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

### ContainerNotFoundException

The specified container was not found for the specified account.

HTTP Status Code: 404

### InternalServerError

The service is temporarily unavailable.

HTTP Status Code: 500

### ObjectNotFoundException

Could not perform an operation on an object that does not exist.

HTTP Status Code: 404

### RequestedRangeNotSatisfiableException

The requested content range is not valid.

HTTP Status Code: 416

## Examples

### Example

The following request downloads the file `mLaw.avi` from the folder `premium/canada` in the container that is identified by the endpoint that is specified in the `Host` header.

```
GET premium/canada/mLaw.avi
Host: aaabbbccdddee.files.mediastore-us-west-2.com
x-amz-Date: 20170323T120000Z
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/us-
west-2/mediastore/aws4_request,SignedHeaders=host;x-amz-date;x-amz-mediastore-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
Content-Length: 0
x-amz-mediastore-version: 2016-07-11
```

## 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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# ListItems

Service: AWS Elemental MediaStore Data Plane

Provides a list of metadata entries about folders and objects in the specified folder.

The `ListItems` response does not include objects that are subject to a transient data rule (objects expire within seconds) in an object lifecycle policy. For more information about rules in object lifecycle policies, see [Components of an Object Lifecycle Policy](#) in the AWS Elemental MediaStore User Guide.

## Request Syntax

```
GET /?MaxResults=MaxResults&NextToken=NextToken&Path=Path HTTP/1.1
```

## URI Request Parameters

The request uses the following URI parameters.

### MaxResults

The maximum number of results to return per API request. For example, you submit a `ListItems` request with `MaxResults` set at 500. Although 2,000 items match your request, the service returns no more than the first 500 items. (The service also returns a `NextToken` value that you can use to fetch the next batch of results.) The service might return fewer results than the `MaxResults` value.

If `MaxResults` is not included in the request, the service defaults to pagination with a maximum of 1,000 results per page.

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

### NextToken

The token that identifies which batch of results that you want to see. For example, you submit a `ListItems` request with `MaxResults` set at 500. The service returns the first batch of results (up to 500) and a `NextToken` value. To see the next batch of results, you can submit the `ListItems` request a second time and specify the `NextToken` value.

Tokens expire after 15 minutes.

## Path

The path in the container from which to retrieve items. Format: <folder name>/<folder name>/<file name>

Length Constraints: Minimum length of 0. Maximum length of 900.

Pattern: `/?(?:[A-Za-z0-9_=\.\-\~]+/){0,10}(?:[A-Za-z0-9_=\.\-\~]+)?/?`

## Request Body

The request does not have a request body.

## Response Syntax

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

{
  "Items": [
    {
      "ContentLength": number,
      "ContentType": "string",
      "ETag": "string",
      "LastModified": number,
      "Name": "string",
      "Type": "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.

### Items

The metadata entries for the folders and objects at the requested path.

Type: Array of [Item](#) objects

### **NextToken**

The token that can be used in a request to view the next set of results. For example, you submit a `ListItems` request that matches 2,000 items with `MaxResults` set at 500. The service returns the first batch of results (up to 500) and a `NextToken` value that can be used to fetch the next batch of results.

Type: String

## **Errors**

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

### **ContainerNotFoundException**

The specified container was not found for the specified account.

HTTP Status Code: 404

### **InternalServerError**

The service is temporarily unavailable.

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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)



- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# PutObject

Service: AWS Elemental MediaStore Data Plane

Uploads an object to the specified path. Object sizes are limited to 25 MB for standard upload availability and 10 MB for streaming upload availability. If the object doesn't fully upload within 30 seconds, MediaStore cancels the request. This applies to all objects, regardless of whether they are uploaded using standard or streaming upload availability.

## Request Syntax

```
PUT /Path+ HTTP/1.1
Content-Type: ContentType
Cache-Control: CacheControl
x-amz-storage-class: StorageClass
x-amz-upload-availability: UploadAvailability
```

*Body*

## URI Request Parameters

The request uses the following URI parameters.

### CacheControl

An optional `CacheControl` header that allows the caller to control the object's cache behavior. Headers can be passed in as specified in the HTTP at <https://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.9>.

Headers with a custom user-defined value are also accepted.

### ContentType

The content type of the object.

Pattern: `^[\\w\\-\\/\\.\\+]{1,255}$`

### Path

The path (including the file name) where the object is stored in the container. Format: `<folder name>/<folder name>/<file name>`

For example, to upload the file `m1aw.avi` to the folder path `premium\\canada` in the container `movies`, enter the path `premium/canada/m1aw.avi`.

Do not include the container name in this path.

If the path includes any folders that don't exist yet, the service creates them. For example, suppose you have an existing `premium/usa` subfolder. If you specify `premium/canada`, the service creates a `canada` subfolder in the `premium` folder. You then have two subfolders, `usa` and `canada`, in the `premium` folder.

There is no correlation between the path to the source and the path (folders) in the container in AWS Elemental MediaStore.

For more information about folders and how they exist in a container, see the [AWS Elemental MediaStore User Guide](#).

The file name is the name that is assigned to the file that you upload. The file can have the same name inside and outside of AWS Elemental MediaStore, or it can have the same name. The file name can include or omit an extension.

Length Constraints: Minimum length of 1. Maximum length of 900.

Pattern: `(?:[A-Za-z0-9_=:\.\\-\~]+/){0,10}[A-Za-z0-9_=:\.\\-\~]+`

Required: Yes

### StorageClass

Indicates the storage class of a Put request. Defaults to high-performance temporal storage class, and objects are persisted into durable storage shortly after being received.

Length Constraints: Minimum length of 1. Maximum length of 16.

Valid Values: STANDARD | TEMPORAL

### UploadAvailability

Indicates the availability of an object while it is still uploading. If the value is set to `streaming`, the object is available for downloading after some initial buffering but before the object is uploaded completely. If the value is set to `standard`, the object is available for downloading only when it is uploaded completely. The default value for this header is `standard`.

To use this header, you must also set the `HTTP Transfer-Encoding` header to `chunked`.

Length Constraints: Minimum length of 1. Maximum length of 16.

Valid Values: STANDARD | STREAMING

## Request Body

The request accepts the following binary data.

### Body

The bytes to be stored.

Required: Yes

## Response Syntax

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

{
  "ContentSHA256": "string",
  "ETag": "string",
  "StorageClass": "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.

### ContentSHA256

The SHA256 digest of the object that is persisted.

Type: String

Length Constraints: Fixed length of 64.

Pattern: [0-9A-Fa-f]{64}

### ETag

Unique identifier of the object in the container.

Type: String

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

Pattern: `[0-9A-Fa-f]+`

### StorageClass

The storage class where the object was persisted. The class should be "Temporal".

Type: String

Length Constraints: Minimum length of 1. Maximum length of 16.

Valid Values: STANDARD | TEMPORAL

## Errors

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

### **ContainerNotFoundException**

The specified container was not found for the specified account.

HTTP Status Code: 404

### **InternalServerError**

The service is temporarily unavailable.

HTTP Status Code: 500

## Examples

### **Regular Upload Request**

The following request puts a file to the container that is identified by the endpoint that is specified in the `Host`: header. It posts it to the folder `premium/canada` in that container and names the file `m1aw.avi`.

```
POST premium/canada/m1aw.avi
Host: aaabbbccdddee.files.mediastore-us-west-2.com
x-amz-Date: 20170323T120000Z
```

```
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/us-  
west-2/mediastore/aws4_request,SignedHeaders=host;x-amz-date;x-amz-mediastore-  
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2  
Content-Length: 0  
x-amz-mediastore-version: 2016-07-11
```

## 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 v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# Data Types

The following data types are supported by AWS Elemental MediaStore:

- [Container](#)
- [CorsRule](#)
- [MetricPolicy](#)
- [MetricPolicyRule](#)
- [Tag](#)

The following data types are supported by AWS Elemental MediaStore Data Plane:

- [Item](#)

## AWS Elemental MediaStore

The following data types are supported by AWS Elemental MediaStore:

- [Container](#)
- [CorsRule](#)
- [MetricPolicy](#)
- [MetricPolicyRule](#)
- [Tag](#)

# Container

Service: AWS Elemental MediaStore

This section describes operations that you can perform on an AWS Elemental MediaStore container.

## Contents

### AccessLoggingEnabled

The state of access logging on the container. This value is `false` by default, indicating that AWS Elemental MediaStore does not send access logs to Amazon CloudWatch Logs. When you enable access logging on the container, MediaStore changes this value to `true`, indicating that the service delivers access logs for objects stored in that container to CloudWatch Logs.

Type: Boolean

Required: No

### ARN

The Amazon Resource Name (ARN) of the container. The ARN has the following format:

`arn:aws:<region>:<account that owns this container>:container/<name of container>`

For example: `arn:aws:mediastore:us-west-2:111122223333:container/movies`

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `arn:aws:mediastore:[a-z]+-[a-z]+-\d:\d{12}:container/[\w-]{1,255}`

Required: No

### CreationTime

Unix timestamp.

Type: Timestamp

Required: No



## Endpoint

The DNS endpoint of the container. Use the endpoint to identify the specific container when sending requests to the data plane. The service assigns this value when the container is created. Once the value has been assigned, it does not change.

Type: String

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

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

Required: No

## Name

The name of the container.

Type: String

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

Pattern: `[\w- ]+`

Required: No

## Status

The status of container creation or deletion. The status is one of the following: `CREATING`, `ACTIVE`, or `DELETING`. While the service is creating the container, the status is `CREATING`. When the endpoint is available, the status changes to `ACTIVE`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 16.

Valid Values: `ACTIVE` | `CREATING` | `DELETING`

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

# CorsRule

Service: AWS Elemental MediaStore

A rule for a CORS policy. You can add up to 100 rules to a CORS policy. If more than one rule applies, the service uses the first applicable rule listed.

## Contents

### AllowedHeaders

Specifies which headers are allowed in a preflight OPTIONS request through the Access-Control-Request-Headers header. Each header name that is specified in Access-Control-Request-Headers must have a corresponding entry in the rule. Only the headers that were requested are sent back.

This element can contain only one wildcard character (\*).

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 100 items.

Length Constraints: Minimum length of 1. Maximum length of 8192.

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

Required: Yes

### AllowedOrigins

One or more response headers that you want users to be able to access from their applications (for example, from a JavaScript XMLHttpRequest object).

Each CORS rule must have at least one AllowedOrigins element. The string value can include only one wildcard character (\*), for example, `http://*.example.com`. Additionally, you can specify only one wildcard character to allow cross-origin access for all origins.

Type: Array of strings

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

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

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

Required: Yes

### **AllowedMethods**

Identifies an HTTP method that the origin that is specified in the rule is allowed to execute.

Each CORS rule must contain at least one AllowedMethods and one AllowedOrigins element.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 4 items.

Valid Values: PUT | GET | DELETE | HEAD

Required: No

### **ExposeHeaders**

One or more headers in the response that you want users to be able to access from their applications (for example, from a JavaScript XMLHttpRequest object).

This element is optional for each rule.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 100 items.

Length Constraints: Minimum length of 1. Maximum length of 8192.

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

Required: No

### **MaxAgeSeconds**

The time in seconds that your browser caches the preflight response for the specified resource.

A CORS rule can have only one MaxAgeSeconds element.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 2147483647.

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

# MetricPolicy

Service: AWS Elemental MediaStore

The metric policy that is associated with the container. A metric policy allows AWS Elemental MediaStore to send metrics to Amazon CloudWatch. In the policy, you must indicate whether you want MediaStore to send container-level metrics. You can also include rules to define groups of objects that you want MediaStore to send object-level metrics for.

To view examples of how to construct a metric policy for your use case, see [Example Metric Policies](#).

## Contents

### ContainerLevelMetrics

A setting to enable or disable metrics at the container level.

Type: String

Valid Values: ENABLED | DISABLED

Required: Yes

### MetricPolicyRules

A parameter that holds an array of rules that enable metrics at the object level. This parameter is optional, but if you choose to include it, you must also include at least one rule. By default, you can include up to five rules. You can also [request a quota increase](#) to allow up to 300 rules per policy.

Type: Array of [MetricPolicyRule](#) objects

Array Members: Minimum number of 1 item. Maximum number of 300 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 Java V2](#)
- [AWS SDK for Ruby V3](#)

# MetricPolicyRule

Service: AWS Elemental MediaStore

A setting that enables metrics at the object level. Each rule contains an object group and an object group name. If the policy includes the MetricPolicyRules parameter, you must include at least one rule. Each metric policy can include up to five rules by default. You can also [request a quota increase](#) to allow up to 300 rules per policy.

## Contents

### ObjectGroup

A path or file name that defines which objects to include in the group. Wildcards (\*) are acceptable.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 900.

Pattern: `/?(?:[A-Za-z0-9_=:\.\\-\~\*]+/){0,10}(?:[A-Za-z0-9_=:\.\\-\~\*]+)?/?`

Required: Yes

### ObjectGroupName

A name that allows you to refer to the object group.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 30.

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

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 Java V2](#)



- [AWS SDK for Ruby V3](#)

# Tag

Service: AWS Elemental MediaStore

A collection of tags associated with a container. Each tag consists of a key:value pair, which can be anything you define. Typically, the tag key represents a category (such as "environment") and the tag value represents a specific value within that category (such as "test," "development," or "production"). You can add up to 50 tags to each container. For more information about tagging, including naming and usage conventions, see [Tagging Resources in MediaStore](#).

## Contents

### Key

Part of the key:value pair that defines a tag. You can use a tag key to describe a category of information, such as "customer." Tag keys are case-sensitive.

Type: String

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

Pattern: `[\p{L}\p{Z}\p{N}_ . : / = + \ - @ ] *`

Required: Yes

### Value

Part of the key:value pair that defines a tag. You can use a tag value to describe a specific value within a category, such as "companyA" or "companyB." Tag values are case-sensitive.

Type: String

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

Pattern: `[\p{L}\p{Z}\p{N}_ . : / = + \ - @ ] *`

Required: No

## See Also

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

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

## AWS Elemental MediaStore Data Plane

The following data types are supported by AWS Elemental MediaStore Data Plane:

- [Item](#)

## Item

Service: AWS Elemental MediaStore Data Plane

A metadata entry for a folder or object.

### Contents

#### ContentLength

The length of the item in bytes.

Type: Long

Valid Range: Minimum value of 0.

Required: No

#### ContentType

The content type of the item.

Type: String

Pattern: `^[\w\-\.\+]{1,255}$`

Required: No

#### ETag

The ETag that represents a unique instance of the item.

Type: String

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

Pattern: `[0-9A-Fa-f]+`

Required: No

#### LastModified

The date and time that the item was last modified.

Type: Timestamp

Required: No

## Name

The name of the item.

Type: String

Pattern: [A-Za-z0-9\_\. \-~]+

Required: No

## Type

The item type (folder or object).

Type: String

Valid Values: OBJECT | FOLDER

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

# Common Parameters

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

## Action

The action to be performed.

Type: string

Required: Yes

## Version

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

Type: string

Required: Yes

## X-Amz-Algorithm

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

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

Type: string

Valid Values: AWS4-HMAC-SHA256

Required: Conditional

## X-Amz-Credential

The credential scope value, which is a string that includes your access key, the date, the region you are targeting, the service you are requesting, and a termination string ("aws4\_request"). The value is expressed in the following format: *access\_key/YYYYMMDD/region/service/aws4\_request*.

For more information, see [Create a signed AWS API request](#) in the *IAM User Guide*.

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

Type: string

Required: Conditional

### **X-Amz-Date**

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

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

Type: string

Required: Conditional

### **X-Amz-Security-Token**

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

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

Type: string

Required: Conditional

### **X-Amz-Signature**

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

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

Type: string

Required: Conditional

### **X-Amz-SignedHeaders**

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

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

Type: string

Required: Conditional



# Common Errors

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

## **AccessDeniedException**

You do not have sufficient access to perform this action.

HTTP Status Code: 400

## **IncompleteSignature**

The request signature does not conform to AWS standards.

HTTP Status Code: 400

## **InternalFailure**

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

HTTP Status Code: 500

## **InvalidAction**

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

HTTP Status Code: 400

## **InvalidClientTokenId**

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

HTTP Status Code: 403

## **NotAuthorized**

You do not have permission to perform this action.

HTTP Status Code: 400

## **OptInRequired**

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

HTTP Status Code: 403

**RequestExpired**

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

HTTP Status Code: 400

**ServiceUnavailable**

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

HTTP Status Code: 503

**ThrottlingException**

The request was denied due to request throttling.

HTTP Status Code: 400

**ValidationError**

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

HTTP Status Code: 400