

---

# Amazon CloudWatch Events

## API Reference

**API Version 2015-10-07**



## **Amazon CloudWatch Events: API Reference**

Copyright © 2016 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
DeleteRule .....	3
Request Syntax .....	3
Request Parameters .....	3
Response Elements .....	3
Errors .....	3
Example .....	3
DescribeRule .....	5
Request Syntax .....	5
Request Parameters .....	5
Response Syntax .....	5
Response Elements .....	5
Errors .....	6
Example .....	6
DisableRule .....	8
Request Syntax .....	8
Request Parameters .....	8
Response Elements .....	8
Errors .....	8
Example .....	8
EnableRule .....	10
Request Syntax .....	10
Request Parameters .....	10
Response Elements .....	10
Errors .....	10
Example .....	10
ListRuleNamesByTarget .....	12
Request Syntax .....	12
Request Parameters .....	12
Response Syntax .....	12
Response Elements .....	12
Errors .....	13
Example .....	13
ListRules .....	15
Request Syntax .....	15
Request Parameters .....	15
Response Syntax .....	15
Response Elements .....	16
Errors .....	16
Example .....	16
ListTargetsByRule .....	18
Request Syntax .....	18
Request Parameters .....	18
Response Syntax .....	18
Response Elements .....	19
Errors .....	19
Example .....	19
PutEvents .....	21
Request Syntax .....	21
Request Parameters .....	21
Response Syntax .....	21
Response Elements .....	21
Errors .....	22
Example .....	22

PutRule .....	24
Request Syntax .....	24
Request Parameters .....	24
Response Syntax .....	25
Response Elements .....	25
Errors .....	25
Example .....	26
PutTargets .....	27
Request Syntax .....	27
Request Parameters .....	27
Response Syntax .....	28
Response Elements .....	28
Errors .....	28
Example .....	28
RemoveTargets .....	30
Request Syntax .....	30
Request Parameters .....	30
Response Syntax .....	30
Response Elements .....	30
Errors .....	31
Example .....	31
TestEventPattern .....	33
Request Syntax .....	33
Request Parameters .....	33
Response Syntax .....	33
Response Elements .....	33
Errors .....	33
Example .....	34
Data Types .....	35
PutEventsRequestEntry .....	36
Contents .....	36
PutEventsResultEntry .....	37
Contents .....	37
PutTargetsResultEntry .....	38
Contents .....	38
RemoveTargetsResultEntry .....	39
Contents .....	39
Rule .....	40
Contents .....	40
Target .....	41
Contents .....	41
Common Parameters .....	42
Common Errors .....	44

# Welcome

---

Amazon CloudWatch Events helps you to respond to state changes in your AWS resources. When your resources change state, they automatically send events into an event stream. You can create rules that match selected events in the stream and route them to targets to take action. You can also use rules to take action on a schedule. For example, you can configure rules to:

- Automatically invoke an AWS Lambda function to update DNS entries when an event notifies you that an EC2 instance entered the running state.
- Direct specific API records from CloudTrail to an Amazon Kinesis stream for detailed analysis of potential security or availability risks.
- Periodically invoke a built-in target to create a snapshot of an Amazon EBS volume.

For more information about CloudWatch Events features, see the [Amazon CloudWatch Events User Guide](#).

Use the following links to get started using the CloudWatch Events Query API:

- [Actions \(p. 2\)](#): An alphabetical list of all CloudWatch Events actions.
- [Data Types \(p. 35\)](#): An alphabetical list of all CloudWatch Events data types.
- [Common Parameters \(p. 42\)](#): Parameters that all Query actions can use.
- [Common Errors \(p. 44\)](#): Client and server errors that all actions can return.
- [Regions and Endpoints](#): Supported regions and endpoints for all AWS products.

Alternatively, you can use one of the [AWS SDKs](#) to access CloudWatch Events using an API tailored to your programming language or platform.

Developers in the AWS developer community also provide their own libraries, which you can find at the following AWS developer centers:

- [Java Developer Center](#)
- [JavaScript Developer Center](#)
- [AWS Mobile Services](#)
- [PHP Developer Center](#)
- [Python Developer Center](#)
- [Ruby Developer Center](#)
- [Windows and .NET Developer Center](#)

# Actions

---

The following actions are supported:

- [DeleteRule](#) (p. 3)
- [DescribeRule](#) (p. 5)
- [DisableRule](#) (p. 8)
- [EnableRule](#) (p. 10)
- [ListRuleNamesByTarget](#) (p. 12)
- [ListRules](#) (p. 15)
- [ListTargetsByRule](#) (p. 18)
- [PutEvents](#) (p. 21)
- [PutRule](#) (p. 24)
- [PutTargets](#) (p. 27)
- [RemoveTargets](#) (p. 30)
- [TestEventPattern](#) (p. 33)

## DeleteRule

Deletes the specified rule.

You must remove all targets from a rule using [RemoveTargets \(p. 30\)](#) before you can delete the rule.

When you delete a rule, incoming events might continue to match to the deleted rule. Please allow a short period of time for changes to take effect.

## Request Syntax

```
{  
  "Name": "string"  
}
```

## Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 42\)](#).

The request accepts the following data in JSON format.

### Name (p. 3)

The name of the rule.

Type: String

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

Pattern: [ \. \- \_A-Za-z0-9 ]+

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 \(p. 44\)](#).

### ConcurrentModificationException

There is concurrent modification on a rule or target.

HTTP Status Code: 400

### InternalException

This exception occurs due to unexpected causes.

HTTP Status Code: 500

## Example

### Deletes a rule named "test"

The following is an example of a DeleteRule request.

### Sample Request

```
POST / HTTP/1.1
```

```
Host: events.<region>.<domain>
x-amz-Date: <Date>
Authorization: AWS4-HMAC-SHA256 Credential=<Credential>,
  SignedHeaders=content-type;date;host;user-agent;x-amz-date;x-amz-target;x-
  amzn-requestid, Signature=<Signature>
User-Agent: <UserAgentString>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Connection: Keep-Alive
X-Amz-Target: AWSEvents.DeleteRule

{
  "Name": "test"
}
```

### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: <RequestId>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Date: <Date>
```



# DescribeRule

Describes the specified rule.

## Request Syntax

```
{  
  "Name": "string"  
}
```

## Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#) (p. 42).

The request accepts the following data in JSON format.

### Name (p. 5)

The name of the rule.

Type: String

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

Pattern: [`\. \- _A-Za-z0-9`]+

Required: Yes

## Response Syntax

```
{  
  "Arn": "string",  
  "Description": "string",  
  "EventPattern": "string",  
  "Name": "string",  
  "RoleArn": "string",  
  "ScheduleExpression": "string",  
  "State": "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.

### Arn (p. 5)

The Amazon Resource Name (ARN) of the rule.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

### Description (p. 5)

The description of the rule.

Type: String

Length Constraints: Maximum length of 512.

### EventPattern (p. 5)

The event pattern.

Type: String

Length Constraints: Maximum length of 2048.

**Name (p. 5)**

The name of the rule.

Type: String

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

Pattern: [\. \- \_A-Za-z0-9 ]+

**RoleArn (p. 5)**

The Amazon Resource Name (ARN) of the IAM role associated with the rule.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

**ScheduleExpression (p. 5)**

The scheduling expression. For example, "cron(0 20 \* \* ? \*)", "rate(5 minutes)".

Type: String

Length Constraints: Maximum length of 256.

**State (p. 5)**

Specifies whether the rule is enabled or disabled.

Type: String

Valid Values: ENABLED | DISABLED

## Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 44\)](#).

**InternalException**

This exception occurs due to unexpected causes.

HTTP Status Code: 500

**ResourceNotFoundException**

The rule does not exist.

HTTP Status Code: 400

## Example

### Describes a rule named "test"

The following is an example of a DescribeRule request and response.

#### Sample Request

```
POST / HTTP/1.1
Host: events.<region>.<domain>
x-amz-Date: <Date>
Authorization: AWS4-HMAC-SHA256 Credential=<Credential>,
  SignedHeaders=content-type;date;host;user-agent;x-amz-date;x-amz-target;x-
  amzn-requestid, Signature=<Signature>
User-Agent: <UserAgentString>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Connection: Keep-Alive
X-Amz-Target: AWSEvents.DescribeRule

{
  "Name": "test"
```

```
}
```

## Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: <RequestId>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Date: <Date>

{
  "Name": "test",
  "EventPattern": "{\"source\":[\"aws.autoscaling\"],\"detail-type\":[\"EC2 Instance Launch Successful\",\"EC2 Instance Terminate Successful\",\"EC2 Instance Launch Unsuccessful\",\"EC2 Instance Terminate Unsuccessful\"]}\",
  "State": "ENABLED",
  "Arn": "arn:aws:events:us-east-1:123456789012:rule/test",
  "Description": "Test rule for Auto Scaling events"
}
```

## DisableRule

Disables the specified rule. A disabled rule won't match any events, and won't self-trigger if it has a schedule expression.

When you disable a rule, incoming events might continue to match to the disabled rule. Please allow a short period of time for changes to take effect.

### Request Syntax

```
{  
  "Name": "string"  
}
```

### Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#) (p. 42).

The request accepts the following data in JSON format.

#### Name (p. 8)

The name of the rule.

Type: String

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

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

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. 44).

#### ConcurrentModificationException

There is concurrent modification on a rule or target.

HTTP Status Code: 400

#### InternalException

This exception occurs due to unexpected causes.

HTTP Status Code: 500

#### ResourceNotFoundException

The rule does not exist.

HTTP Status Code: 400

### Example

#### Disables a rule named "test"

The following is an example of a DisableRule request.

## Sample Request

```
POST / HTTP/1.1
Host: events.<region>.<domain>
x-amz-Date: <Date>
Authorization: AWS4-HMAC-SHA256 Credential=<Credential>,
  SignedHeaders=content-type;date;host;user-agent;x-amz-date;x-amz-target;x-
  amzn-requestid, Signature=<Signature>
User-Agent: <UserAgentString>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Connection: Keep-Alive
X-Amz-Target: AWSEvents.DisableRule

{
  "Name": "test"
}
```

## Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: <RequestId>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Date: <Date>
```

## EnableRule

Enables the specified rule. If the rule does not exist, the operation fails.

When you enable a rule, incoming events might not immediately start matching to a newly enabled rule. Please allow a short period of time for changes to take effect.

### Request Syntax

```
{  
  "Name": "string"  
}
```

### Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#) (p. 42).

The request accepts the following data in JSON format.

#### **Name** (p. 10)

The name of the rule.

Type: String

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

Pattern: [ \ . \ \_ \_ A - Z a - z 0 - 9 ] +

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. 44).

#### **ConcurrentModificationException**

There is concurrent modification on a rule or target.

HTTP Status Code: 400

#### **InternalException**

This exception occurs due to unexpected causes.

HTTP Status Code: 500

#### **ResourceNotFoundException**

The rule does not exist.

HTTP Status Code: 400

### Example

#### Enables a rule named "test"

The following is an example of an EnableRule request.

#### Sample Request

```
POST / HTTP/1.1
```

```
Host: events.<region>.<domain>
x-amz-Date: <Date>
Authorization: AWS4-HMAC-SHA256 Credential=<Credential>,
  SignedHeaders=content-type;date;host;user-agent;x-amz-date;x-amz-target;x-
  amzn-requestid, Signature=<Signature>
User-Agent: <UserAgentString>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Connection: Keep-Alive
X-Amz-Target: AWSEvents.EnableRule

{
  "Name": "test"
}
```

### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: <RequestId>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Date: <Date>
```

## ListRuleNamesByTarget

Lists the rules for the specified target. You can see which of the rules in Amazon CloudWatch Events can invoke a specific target in your account.

### Request Syntax

```
{  
  "Limit": number,  
  "NextToken": "string",  
  "TargetArn": "string"  
}
```

### Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#) (p. 42).

The request accepts the following data in JSON format.

#### Limit (p. 12)

The maximum number of results to return.

Type: Integer

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

Required: No

#### NextToken (p. 12)

The token returned by a previous call to retrieve the next set of results.

Type: String

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

Required: No

#### TargetArn (p. 12)

The Amazon Resource Name (ARN) of the target resource.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Required: Yes

### Response Syntax

```
{  
  "NextToken": "string",  
  "RuleNames": [ "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. 12)

Indicates whether there are additional results to retrieve. If there are no more results, the value is null.



Type: String

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

#### **RuleNames** (p. 12)

The names of the rules that can invoke the given target.

Type: array of Strings

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

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

## Errors

For information about the errors that are common to all actions, see [Common Errors](#) (p. 44).

### **InternalException**

This exception occurs due to unexpected causes.

HTTP Status Code: 500

## Example

### Lists rule names by target with the specified ARN

The following is an example of a `ListRuleNamesByTarget` request and response.

#### Sample Request

```
POST / HTTP/1.1
Host: events.<region>.<domain>
x-amz-Date: <Date>
Authorization: AWS4-HMAC-SHA256 Credential=<Credential>,
  SignedHeaders=content-type;date;host;user-agent;x-amz-date;x-amz-target;x-
  amzn-requestid, Signature=<Signature>
User-Agent: <UserAgentString>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Connection: Keep-Alive
X-Amz-Target: AWSEvents.ListRuleNamesByTarget

{
  "TargetArn": "arn:aws:lambda:us-
  east-1:123456789012:function:MyFunction",
  "NextToken": "",
  "Limit": 0
}
```

#### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: <RequestId>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Date: <Date>

{
  "RuleNames": [
    "test1",
```

```
    "test2",  
    "test3",  
    "test4",  
    "test5"  
  ]  
}
```

## ListRules

Lists your Amazon CloudWatch Events rules. You can either list all the rules or you can provide a prefix to match to the rule names.

### Request Syntax

```
{
  "Limit": number,
  "NamePrefix": "string",
  "NextToken": "string"
}
```

### Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#) (p. 42).

The request accepts the following data in JSON format.

#### Limit (p. 15)

The maximum number of results to return.

Type: Integer

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

Required: No

#### NamePrefix (p. 15)

The prefix matching the rule name.

Type: String

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

Pattern: [`\. \_ - _A - Za - z0 - 9` ]+

Required: No

#### NextToken (p. 15)

The token returned by a previous call to retrieve the next set of results.

Type: String

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

Required: No

### Response Syntax

```
{
  "NextToken": "string",
  "Rules": [
    {
      "Arn": "string",
      "Description": "string",
      "EventPattern": "string",
      "Name": "string",
      "RoleArn": "string",
      "ScheduleExpression": "string",
      "State": "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. 15)

Indicates whether there are additional results to retrieve. If there are no more results, the value is null.

Type: String

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

### Rules (p. 15)

The rules that match the specified criteria.

Type: array of [Rule \(p. 40\)](#) objects

## Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 44\)](#).

### InternalException

This exception occurs due to unexpected causes.

HTTP Status Code: 500

## Example

### Lists all the rules that start with the letter "t" with a page size of 1

The following is an example of a ListRules request and response.

#### Sample Request

```
POST / HTTP/1.1
Host: events.<region>.<domain>
x-amz-Date: <Date>
Authorization: AWS4-HMAC-SHA256 Credential=<Credential>,
  SignedHeaders=content-type;date;host;user-agent;x-amz-date;x-amz-target;x-
  amzn-requestid, Signature=<Signature>
User-Agent: <UserAgentString>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Connection: Keep-Alive
X-Amz-Target: AWSEvents.ListRules

{
  "NamePrefix": "t",
  "Limit": 1
}
```

#### Sample Response

```
HTTP/1.1 200 OK
```

Amazon CloudWatch Events API Reference  
Example

---

```
x-amzn-RequestId: <RequestId>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Date: <Date>

{
  "Rules": [
    {
      "EventPattern": "{\"source\":[\"aws.autoscaling\"],\"detail-type\":[\"EC2 Instance Launch Successful\",\"EC2 Instance Terminate Successful\",\"EC2 Instance Launch Unsuccessful\",\"EC2 Instance Terminate Unsuccessful\"]}\",
      "State": "DISABLED",
      "Name": "test",
      "Arn": "arn:aws:events:us-east-1:123456789012:rule/test",
      "Description": "Test rule for Auto Scaling events"
    }
  ],
  "NextToken": "ABCDEgAAAAAAAAAAQAAABCXtD8i7XlyFv5XFKH8GrudAAAAQIoQ0+7qXp63vQf1pvVklfHFd+p2QgY36pjlAqsSsrkNbOtTePaCeJqN80+jbu66UhpJh7huA9r0iY9zjdtZ3vsAAAAGAAAAAAAAAF5MZWktllmMuLd9gUjryM4sL9EG5IkcPUm60Vq1tzyYw=="
}
```

## ListTargetsByRule

Lists the targets assigned to the specified rule.

### Request Syntax

```
{  
  "Limit": number,  
  "NextToken": "string",  
  "Rule": "string"  
}
```

### Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#) (p. 42).

The request accepts the following data in JSON format.

#### Limit (p. 18)

The maximum number of results to return.

Type: Integer

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

Required: No

#### NextToken (p. 18)

The token returned by a previous call to retrieve the next set of results.

Type: String

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

Required: No

#### Rule (p. 18)

The name of the rule.

Type: String

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

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

Required: Yes

### Response Syntax

```
{  
  "NextToken": "string",  
  "Targets": [  
    {  
      "Arn": "string",  
      "Id": "string",  
      "Input": "string",  
      "InputPath": "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. 18)

Indicates whether there are additional results to retrieve. If there are no more results, the value is null.

Type: String

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

### Targets (p. 18)

The targets assigned to the rule.

Type: array of [Target \(p. 41\)](#) objects

## Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 44\)](#).

### InternalException

This exception occurs due to unexpected causes.

HTTP Status Code: 500

### ResourceNotFoundException

The rule does not exist.

HTTP Status Code: 400

## Example

### Lists the targets associated with a rule named "test"

The following is an example of a ListTargetsByRule request and response.

#### Sample Request

```
POST / HTTP/1.1
Host: events.<region>.<domain>
x-amz-Date: <Date>
Authorization: AWS4-HMAC-SHA256 Credential=<Credential>,
  SignedHeaders=content-type;date;host;user-agent;x-amz-date;x-amz-target;x-
  amzn-requestid, Signature=<Signature>
User-Agent: <UserAgentString>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Connection: Keep-Alive
X-Amz-Target: AWSEvents.ListTargetsByRule

{
  "Rule": "test"
}
```

#### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: <RequestId>
```

```
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Date: <Date>

{
  "Targets": [
    {
      "Id": "MyTargetId",
      "Arn": "arn:aws:lambda:us-
east-1:123456789012:function:MyFunction"
    }
  ]
}
```



## PutEvents

Sends custom events to Amazon CloudWatch Events so that they can be matched to rules.

### Request Syntax

```
{
  "Entries": [
    {
      "Detail": "string",
      "DetailType": "string",
      "Resources": [ "string" ],
      "Source": "string",
      "Time": number
    }
  ]
}
```

### Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#) (p. 42).

The request accepts the following data in JSON format.

#### Entries (p. 21)

The entry that defines an event in your system. You can specify several parameters for the entry such as the source and type of the event, resources associated with the event, and so on.

Type: array of [PutEventsRequestEntry](#) (p. 36) objects

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

Required: Yes

### Response Syntax

```
{
  "Entries": [
    {
      "ErrorCode": "string",
      "ErrorMessage": "string",
      "EventId": "string"
    }
  ],
  "FailedEntryCount": 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.

#### Entries (p. 21)

The successfully and unsuccessfully ingested events results. If the ingestion was successful, the entry will have the event ID in it. Otherwise, you can use the error code and error message to identify the problem with the entry.

Type: array of [PutEventsResultEntry](#) (p. 37) objects

**FailedEntryCount** (p. 21)

The number of failed entries.

Type: Integer

## Errors

For information about the errors that are common to all actions, see [Common Errors](#) (p. 44).

**InternalException**

This exception occurs due to unexpected causes.

HTTP Status Code: 500

## Example

### Sends two custom events

The following is an example of a PutEvents request and response.

#### Sample Request

```
POST / HTTP/1.1
Host: events.<region>.<domain>
x-amz-Date: <Date>
Authorization: AWS4-HMAC-SHA256 Credential=<Credential>,
  SignedHeaders=content-type;date;host;user-agent;x-amz-date;x-amz-target;x-
  amzn-requestid, Signature=<Signature>
User-Agent: <UserAgentString>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Connection: Keep-Alive
X-Amz-Target: AWSEvents.PutEvents

{
  "Entries": [
    {
      "Source": "com.mycompany.myapp",
      "Detail": { "key1": "value1", "key2": "value2" },
      "Resources": [
        "resource1",
        "resource2"
      ],
      "DetailType": "myDetailType"
    },
    {
      "Source": "com.mycompany.myapp",
      "Detail": { "key1": "value3", "key2": "value4" },
      "Resources": [
        "resource1",
        "resource2"
      ],
      "DetailType": "myDetailType"
    }
  ]
}
```

## Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: <RequestId>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Date: <Date>

{
  "FailedEntryCount": 0,
  "Entries": [
    {
      "EventId": "11710aed-b79e-4468-a20b-bb3c0c3b4860"
    },
    {
      "EventId": "d804d26a-88db-4b66-9eaf-9a11c708ae82"
    }
  ]
}
```

## PutRule

Creates or updates the specified rule. Rules are enabled by default, or based on value of the state. You can disable a rule using [DisableRule \(p. 8\)](#).

When you create or update a rule, incoming events might not immediately start matching to new or updated rules. Please allow a short period of time for changes to take effect.

A rule must contain at least an EventPattern or ScheduleExpression. Rules with EventPatterns are triggered when a matching event is observed. Rules with ScheduleExpressions self-trigger based on the given schedule. A rule can have both an EventPattern and a ScheduleExpression, in which case the rule will trigger on matching events as well as on a schedule.

Most services in AWS treat : or / as the same character in Amazon Resource Names (ARNs). However, CloudWatch Events uses an exact match in event patterns and rules. Be sure to use the correct ARN characters when creating event patterns so that they match the ARN syntax in the event you want to match.

## Request Syntax

```
{
  "Description": "string",
  "EventPattern": "string",
  "Name": "string",
  "RoleArn": "string",
  "ScheduleExpression": "string",
  "State": "string"
}
```

## Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 42\)](#).

The request accepts the following data in JSON format.

### Description (p. 24)

A description of the rule.  
Type: String  
Length Constraints: Maximum length of 512.  
Required: No

### EventPattern (p. 24)

The event pattern.  
Type: String  
Length Constraints: Maximum length of 2048.  
Required: No

### Name (p. 24)

The name of the rule that you are creating or updating.  
Type: String  
Length Constraints: Minimum length of 1. Maximum length of 64.  
Pattern: [ \ . \ \_ \_ A - Z a - z 0 - 9 ] +  
Required: Yes

### RoleArn (p. 24)

The Amazon Resource Name (ARN) of the IAM role associated with the rule.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Required: No

#### **ScheduleExpression** (p. 24)

The scheduling expression. For example, "cron(0 20 \* \* ? \*)", "rate(5 minutes)".

Type: String

Length Constraints: Maximum length of 256.

Required: No

#### **State** (p. 24)

Indicates whether the rule is enabled or disabled.

Type: String

Valid Values: `ENABLED` | `DISABLED`

Required: No

## Response Syntax

```
{  
  "RuleArn": "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.

#### **RuleArn** (p. 25)

The Amazon Resource Name (ARN) of the rule.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

## Errors

For information about the errors that are common to all actions, see [Common Errors](#) (p. 44).

#### **ConcurrentModificationException**

There is concurrent modification on a rule or target.

HTTP Status Code: 400

#### **InternalException**

This exception occurs due to unexpected causes.

HTTP Status Code: 500

#### **InvalidEventPatternException**

The event pattern is not valid.

HTTP Status Code: 400

#### **LimitExceededException**

You tried to create more rules or add more targets to a rule than is allowed.

HTTP Status Code: 400

## Example

### Creates a rule named "test" that matches events from Amazon EC2

The following is an example of a PutRule request and response.

#### Sample Request

```
POST / HTTP/1.1
Host: events.<region>.<domain>
x-amz-Date: <Date>
Authorization: AWS4-HMAC-SHA256 Credential=<Credential>,
  SignedHeaders=content-type;date;host;user-agent;x-amz-date;x-amz-target;x-
  amzn-requestid, Signature=<Signature>
User-Agent: <UserAgentString>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Connection: Keep-Alive
X-Amz-Target: AWSEvents.PutRule

{
  "Name": "test",
  "EventPattern": "{ \"source\": [\"aws.ec2\"] }"
}
```

#### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: <RequestId>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Date: <Date>

{
  "RuleArn": "arn:aws:events:us-east-1:123456789012:rule/test"
}
```

## PutTargets

Adds the specified targets to the specified rule. Updates the targets if they are already associated with the role.

Targets are the resources that can be invoked when a rule is triggered. Example targets include AWS Lambda functions, Amazon Kinesis streams, and built-in targets. Note that creating rules with built-in targets is supported only in the AWS Management Console.

In order to be able to make API calls against the resources you own, Amazon CloudWatch Events needs the appropriate permissions. For AWS Lambda and Amazon SNS resources, CloudWatch Events relies on resource-based policies. For Amazon Kinesis streams, CloudWatch Events relies on IAM roles. For more information, see [Authentication and Access Control](#) in the *Amazon CloudWatch Events User Guide*.

**Input** and **InputPath** are mutually-exclusive and optional parameters of a target. When a rule is triggered due to a matched event, if for a target:

- Neither **Input** nor **InputPath** is specified, then the entire event is passed to the target in JSON form.
- **InputPath** is specified in the form of JSONPath (for example, `$.detail`), then only the part of the event specified in the path is passed to the target (for example, only the detail part of the event is passed).
- **Input** is specified in the form of a valid JSON, then the matched event is overridden with this constant.

When you add targets to a rule, when the associated rule triggers, new or updated targets might not be immediately invoked. Please allow a short period of time for changes to take effect.

## Request Syntax

```
{
  "Rule": "string",
  "Targets": [
    {
      "Arn": "string",
      "Id": "string",
      "Input": "string",
      "InputPath": "string"
    }
  ]
}
```

## Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#) (p. 42).

The request accepts the following data in JSON format.

### Rule (p. 27)

The name of the rule.

Type: String

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

Pattern: `[\.\_\_A-Za-z0-9]+`

Required: Yes

### Targets (p. 27)

The targets to update or add to the rule.

Type: array of [Target \(p. 41\)](#) objects

Required: Yes

## Response Syntax

```
{
  "FailedEntries": [
    {
      "ErrorCode": "string",
      "ErrorMessage": "string",
      "TargetId": "string"
    }
  ],
  "FailedEntryCount": 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.

### **FailedEntries (p. 28)**

The failed target entries.

Type: array of [PutTargetsResultEntry \(p. 38\)](#) objects

### **FailedEntryCount (p. 28)**

The number of failed entries.

Type: Integer

## Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 44\)](#).

### **ConcurrentModificationException**

There is concurrent modification on a rule or target.

HTTP Status Code: 400

### **InternalException**

This exception occurs due to unexpected causes.

HTTP Status Code: 500

### **LimitExceededException**

You tried to create more rules or add more targets to a rule than is allowed.

HTTP Status Code: 400

### **ResourceNotFoundException**

The rule does not exist.

HTTP Status Code: 400

## Example

### Adds a target to a Lambda function with the ID "MyTargetId" to the rule named "test"

The following is an example of a PutTargets request.



## Sample Request

```
POST / HTTP/1.1
Host: events.<region>.<domain>
x-amz-Date: <Date>
Authorization: AWS4-HMAC-SHA256 Credential=<Credential>,
  SignedHeaders=content-type;date;host;user-agent;x-amz-date;x-amz-target;x-
  amzn-requestid, Signature=<Signature>
User-Agent: <UserAgentString>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Connection: Keep-Alive
X-Amz-Target: AWSEvents.PutTargets

{
  "Rule": "test",
  "Targets": [
    {
      "Id": "MyTargetId",
      "Arn": "arn:aws:lambda:us-
east-1:123456789012:function:MyFunction"
    }
  ]
}
```

## Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: <RequestId>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Date: <Date>

{
  "FailedEntries": [],
  "FailedEntryCount": 0
}
```

## RemoveTargets

Removes the specified targets from the specified rule. When the rule is triggered, those targets are no longer be invoked.

When you remove a target, when the associated rule triggers, removed targets might continue to be invoked. Please allow a short period of time for changes to take effect.

### Request Syntax

```
{
  "Ids": [ "string" ],
  "Rule": "string"
}
```

### Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#) (p. 42).

The request accepts the following data in JSON format.

#### Ids (p. 30)

The IDs of the targets to remove from the rule.

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 64.

Pattern: [`\.\-_A-Za-z0-9`]+

Required: Yes

#### Rule (p. 30)

The name of the rule.

Type: String

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

Pattern: [`\.\-_A-Za-z0-9`]+

Required: Yes

### Response Syntax

```
{
  "FailedEntries": [
    {
      "ErrorCode": "string",
      "ErrorMessage": "string",
      "TargetId": "string"
    }
  ],
  "FailedEntryCount": 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.

### **FailedEntries (p. 30)**

The failed target entries.

Type: array of [RemoveTargetsResultEntry \(p. 39\)](#) objects

### **FailedEntryCount (p. 30)**

The number of failed entries.

Type: Integer

## Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 44\)](#).

### **ConcurrentModificationException**

There is concurrent modification on a rule or target.

HTTP Status Code: 400

### **InternalException**

This exception occurs due to unexpected causes.

HTTP Status Code: 500

### **ResourceNotFoundException**

The rule does not exist.

HTTP Status Code: 400

## Example

### Removes a target with ID "MyTargetId" from a rule named "test"

The following is an example of a `RemoveTargets` request.

#### Sample Request

```
POST / HTTP/1.1
Host: events.<region>.<domain>
x-amz-Date: <Date>
Authorization: AWS4-HMAC-SHA256 Credential=<Credential>,
  SignedHeaders=content-type;date;host;user-agent;x-amz-date;x-amz-target;x-
  amzn-requestid, Signature=<Signature>
User-Agent: <UserAgentString>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Connection: Keep-Alive
X-Amz-Target: AWSEvents.RemoveTargets

{
  "Rule": "test",
  "Ids": [
    "MyTargetId"
  ]
}
```

#### Sample Response

```
HTTP/1.1 200 OK
```

```
x-amzn-RequestId: <RequestId>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Date: <Date>

{
  "FailedEntries": [],
  "FailedEntryCount": 0
}
```

# TestEventPattern

Tests whether the specified event pattern matches the provided event.

Most services in AWS treat `:` or `/` as the same character in Amazon Resource Names (ARNs).

However, CloudWatch Events uses an exact match in event patterns and rules. Be sure to use the correct ARN characters when creating event patterns so that they match the ARN syntax in the event you want to match.

## Request Syntax

```
{
  "Event": "string",
  "EventPattern": "string"
}
```

## Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters \(p. 42\)](#).

The request accepts the following data in JSON format.

### Event (p. 33)

The event, in JSON format, to test against the event pattern.

Type: String

Required: Yes

### EventPattern (p. 33)

The event pattern.

Type: String

Length Constraints: Maximum length of 2048.

Required: Yes

## Response Syntax

```
{
  "Result": boolean
}
```

## Response Elements

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

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

### Result (p. 33)

Indicates whether the event matches the event pattern.

Type: Boolean

## Errors

For information about the errors that are common to all actions, see [Common Errors \(p. 44\)](#).

**InternalException**

This exception occurs due to unexpected causes.

HTTP Status Code: 500

**InvalidEventPatternException**

The event pattern is not valid.

HTTP Status Code: 400

## Example

### Tests that a given event matches a given event pattern

The following is an example of a TestEventPattern request and response.

#### Sample Request

```
POST / HTTP/1.1
Host: events.<region>.<domain>
x-amz-Date: <Date>
Authorization: AWS4-HMAC-SHA256 Credential=<Credential>,
  SignedHeaders=content-type;date;host;user-agent;x-amz-date;x-amz-target;x-
  amzn-requestid, Signature=<Signature>
User-Agent: <UserAgentString>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Connection: Keep-Alive
X-Amz-Target: AWSEvents.TestEventPattern

{
  "EventPattern": "{\"source\": [\"com.mycompany.myapp\"]}\",
  "Event": "{\"id\": \"e00c66cb-fe7a-4fcc-81ad-58eb60f5d96b\", \"detail-
  type\": \"myDetailType\", \"source\": \"com.mycompany.myapp\", \"account
  \": \"123456789012\", \"time\": \"2016-01-10T01:29:23Z\", \"region\":
  \"us-east-1\", \"resources\": [\"resource1\", \"resource2\"], \"detail\":
  {\"key1\": \"value1\", \"key2\": \"value2\"}}\"
}
```

#### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: <RequestId>
Content-Type: application/x-amz-json-1.1
Content-Length: <PayloadSizeBytes>
Date: <Date>

{
  "Result": true
}
```

# Data Types

---

The Amazon CloudWatch Events 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:

- [PutEventsRequestEntry](#) (p. 36)
- [PutEventsResultEntry](#) (p. 37)
- [PutTargetsResultEntry](#) (p. 38)
- [RemoveTargetsResultEntry](#) (p. 39)
- [Rule](#) (p. 40)
- [Target](#) (p. 41)

# PutEventsRequestEntry

Represents an event to be submitted.

## Contents

### Detail

In the JSON sense, an object containing fields, which may also contain nested sub-objects. No constraints are imposed on its contents.

Type: String

Required: No

### DetailType

Free-form string used to decide what fields to expect in the event detail.

Type: String

Required: No

### Resources

AWS resources, identified by Amazon Resource Name (ARN), which the event primarily concerns. Any number, including zero, may be present.

Type: array of Strings

Required: No

### Source

The source of the event.

Type: String

Required: No

### Time

The timestamp of the event, per [RFC3339](#). If no timestamp is provided, the timestamp of the [PutEvents \(p. 21\)](#) call will be used.

Type: Timestamp

Required: No



## PutEventsResultEntry

Represents an event that failed to be submitted.

### Contents

**ErrorCode**

The error code that indicates why the event submission failed.

Type: String

Required: No

**ErrorMessage**

The error message that explains why the event submission failed.

Type: String

Required: No

**EventId**

The ID of the event.

Type: String

Required: No

# PutTargetsResultEntry

Represents a target that failed to be added to a rule.

## Contents

### **ErrorCode**

The error code that indicates why the target addition failed.

Type: String

Required: No

### **ErrorMessage**

The error message that explains why the target addition failed.

Type: String

Required: No

### **TargetId**

The ID of the target.

Type: String

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

Pattern: [ \. \\_ - A - Z a - z 0 - 9 ] +

Required: No

# RemoveTargetsResultEntry

Represents a target that failed to be removed from a rule.

## Contents

### **ErrorCode**

The error code that indicates why the target removal failed.

Type: String

Required: No

### **ErrorMessage**

The error message that explains why the target removal failed.

Type: String

Required: No

### **TargetId**

The ID of the target.

Type: String

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

Pattern: [ \. \\_ A-Z a-z 0-9 ]+

Required: No

# Rule

Contains information about a rule in Amazon CloudWatch Events.

## Contents

### **Arn**

The Amazon Resource Name (ARN) of the rule.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Required: No

### **Description**

The description of the rule.

Type: String

Length Constraints: Maximum length of 512.

Required: No

### **EventPattern**

The event pattern of the rule.

Type: String

Length Constraints: Maximum length of 2048.

Required: No

### **Name**

The name of the rule.

Type: String

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

Pattern: [ \ . \ \_ \_ A - Z a - z 0 - 9 ] +

Required: No

### **RoleArn**

The Amazon Resource Name (ARN) of the role that is used for target invocation.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Required: No

### **ScheduleExpression**

The scheduling expression. For example, "cron(0 20 \* \* ? \*)", "rate(5 minutes)".

Type: String

Length Constraints: Maximum length of 256.

Required: No

### **State**

The state of the rule.

Type: String

Valid Values: `ENABLED` | `DISABLED`

Required: No

## Target

Targets are the resources that can be invoked when a rule is triggered. Target types include AWS Lambda functions, Amazon Kinesis streams, and built-in targets.

**Input** and **InputPath** are mutually-exclusive and optional parameters of a target. When a rule is triggered due to a matched event, if for a target:

- Neither **Input** nor **InputPath** is specified, then the entire event is passed to the target in JSON form.
- **InputPath** is specified in the form of JSONPath (for example, `$.detail`), then only the part of the event specified in the path is passed to the target (for example, only the detail part of the event is passed).
- **Input** is specified in the form of a valid JSON, then the matched event is overridden with this constant.

## Contents

### Arn

The Amazon Resource Name (ARN) of the target.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Required: Yes

### Id

The ID of the target.

Type: String

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

Pattern: `[ \. \_ - _A-Za-z0-9 ]+`

Required: Yes

### Input

Valid JSON text passed to the target. For more information, see [The JavaScript Object Notation \(JSON\) Data Interchange Format](#).

Type: String

Length Constraints: Maximum length of 8192.

Required: No

### InputPath

The value of the JSONPath that is used for extracting part of the matched event when passing it to the target. For more information about JSON paths, see [JSONPath](#).

Type: String

Length Constraints: Maximum length of 256.

Required: No

# Common Parameters

---

The following table lists the parameters that all actions use for signing Signature Version 4 requests. Any action-specific parameters are listed in the topic for that action. To view sample requests, see [Examples of Signed Signature Version 4 Requests](#) or [Signature Version 4 Test Suite](#) in the *Amazon Web Services General Reference*.

**Action**

The action to be performed.

Type: string

Required: Yes

**Version**

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

Type: string

Required: Yes

**X-Amz-Algorithm**

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

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

Type: string

Valid Values: `AWS4-HMAC-SHA256`

Required: Conditional

**X-Amz-Credential**

The credential scope value, which is a string that includes your access key, the date, the region you are targeting, the service you are requesting, and a termination string ("aws4\_request"). The value is expressed in the following format: `access_key/YYYYMMDD/region/service/aws4_request`.

For more information, see [Task 2: Create a String to Sign for Signature Version 4](#) in the *Amazon Web Services General Reference*.

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

Type: string

Required: Conditional

**X-Amz-Date**

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

Condition: X-Amz-Date is optional for all requests; it can be used to override the date used for signing requests. If the Date header is specified in the ISO 8601 basic format, X-Amz-Date is not required. When X-Amz-Date is used, it always overrides the value of the Date header. For more information, see [Handling Dates in Signature Version 4](#) in the *Amazon Web Services General Reference*.

Type: string

Required: Conditional

**X-Amz-Security-Token**

The temporary security token that was obtained through a call to AWS Security Token Service. For a list of services that support AWS Security Token Service, go to [Using Temporary Security Credentials to Access AWS](#) in *Using Temporary Security Credentials*.

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 common errors that all actions return. Any action-specific errors are listed in the topic for the action.

**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

**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

**Throttling**

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