AWS Config

API Reference

API Version 2014-11-12

AWS Config: API Reference

Table of Contents

Welcome

AWS Config provides a way to keep track of the configurations of all the AWS resources associated with your AWS account. You can use AWS Config to get the current and historical configurations of each AWS resource and also to get information about the relationship between the resources. An AWS resource can be an Amazon Compute Cloud (Amazon EC2) instance, an Elastic Block Store (EBS) volume, an Elastic network Interface (ENI), or a security group. For a complete list of resources currently supported by AWS Config, see Supported AWS Resources.

You can access and manage AWS Config through the AWS Management Console, the AWS Command Line Interface (AWS CLI), the AWS Config API, or the AWS SDKs for AWS Config

This reference guide contains documentation for the AWS Config API and the AWS CLI commands that you can use to manage AWS Config.

The AWS Config API uses the Signature Version 4 protocol for signing requests. For more information about how to sign a request with this protocol, see Signature Version 4 Signing Process.

For detailed information about AWS Config features and their associated actions or commands, as well as how to work with AWS Management Console, see What Is AWS Config? in the *AWS Config Developer Guide*.

This document was last updated on April 7, 2016.

Actions

The following actions are supported:

DeleteConfigRule

Deletes the specified AWS Config rule and all of its evaluation results.

AWS Config sets the state of a rule to `DELETING` until the deletion is complete. You cannot update a rule while it is in this state. If you make a `PutConfigRule` or `DeleteConfigRule` request for the rule, you will receive a `ResourceInUseException`.

You can check the state of a rule by using the `DescribeConfigRules` request.

Request Syntax

```
{
    "ConfigRuleName": "string"
}
```

Request Parameters

For information about the common parameters that all actions use, see Common Parameters (p. 75).

The request requires the following data in JSON format.

ConfigRuleName
    The name of the AWS Config rule that you want to delete.

    Type: String

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

    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. 79).

NoSuchConfigRuleException
    One or more AWS Config rules in the request are invalid. Verify that the rule names are correct and try again.

    HTTP Status Code: 400

ResourceInUseException
    The rule is currently being deleted. Wait for a while and try again.

    HTTP Status Code: 400

DeleteDeliveryChannel

Deletes the specified delivery channel.

The delivery channel cannot be deleted if it is the only delivery channel and the configuration recorder is still running. To delete the delivery channel, stop the running configuration recorder using the StopConfigurationRecorder (p. 50) action.

Request Syntax

```
{
    "DeliveryChannelName": "string"
}
```

Request Parameters

For information about the common parameters that all actions use, see Common Parameters (p. 75).

The request requires the following data in JSON format.

DeliveryChannelName
    The name of the delivery channel to delete.

    Type: String

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

    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. 79).

LastDeliveryChannelDeleteFailedException
    You cannot delete the delivery channel you specified because the configuration recorder is running.

    HTTP Status Code: 400

NoSuchDeliveryChannelException
    You have specified a delivery channel that does not exist.

    HTTP Status Code: 400

DeliverConfigSnapshot

Schedules delivery of a configuration snapshot to the Amazon S3 bucket in the specified delivery channel. After the delivery has started, AWS Config sends following notifications using an Amazon SNS topic that you have specified.

- Notification of starting the delivery.
- Notification of delivery completed, if the delivery was successfully completed.
- Notification of delivery failure, if the delivery failed to complete.

Request Syntax

```
{
    "deliveryChannelName": "string"
}
```

Request Parameters

For information about the common parameters that all actions use, see Common Parameters (p. 75).

The request requires the following data in JSON format.

deliveryChannelName
> The name of the delivery channel through which the snapshot is delivered.
>
> Type: String
>
> Length constraints: Minimum length of 1. Maximum length of 256.
>
> Required: Yes

Response Syntax

```
{
    "configSnapshotId": "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.

configSnapshotId
> The ID of the snapshot that is being created.
>
> Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 79).

NoAvailableConfigurationRecorderException

There are no configuration recorders available to provide the role needed to describe your resources. Create a configuration recorder.

HTTP Status Code: 400

NoRunningConfigurationRecorderException

There is no configuration recorder running.

HTTP Status Code: 400

NoSuchDeliveryChannelException

You have specified a delivery channel that does not exist.

HTTP Status Code: 400

DescribeComplianceByConfigRule

Indicates whether the specified AWS Config rules are compliant. If a rule is noncompliant, this action returns the number of AWS resources that do not comply with the rule.

A rule is compliant if all of the evaluated resources comply with it, and it is noncompliant if any of these resources do not comply.

If AWS Config has no current evaluation results for the rule, it returns `INSUFFICIENT_DATA`. This result might indicate one of the following conditions:

* AWS Config has never invoked an evaluation for the rule. To check whether it has, use the `DescribeConfigRuleEvaluationStatus` action to get the `LastSuccessfulInvocationTime` and `LastFailedInvocationTime`.
* The rule's AWS Lambda function is failing to send evaluation results to AWS Config. Verify that the role that you assigned to your configuration recorder includes the `config:PutEvaluations` permission. If the rule is a customer managed rule, verify that the AWS Lambda execution role includes the `config:PutEvaluations` permission.
* The rule's AWS Lambda function has returned `NOT_APPLICABLE` for all evaluation results. This can occur if the resources were deleted or removed from the rule's scope.

Request Syntax

```
{
    "ComplianceTypes": [
        "string"
    ],
    "ConfigRuleNames": [
        "string"
    ],
    "NextToken": "string"
}
```

Request Parameters

For information about the common parameters that all actions use, see Common Parameters (p. 75).

The request requires the following data in JSON format.

ComplianceTypes
Filters the results by compliance.

The allowed values are `COMPLIANT`, `NON_COMPLIANT`, and `INSUFFICIENT_DATA`.

Type: array of Strings

Length constraints: Minimum of 0 item(s) in the list. Maximum of 3 item(s) in the list.

Required: No

ConfigRuleNames
Specify one or more AWS Config rule names to filter the results by rule.

Type: array of Strings

Length constraints: Minimum of 0 item(s) in the list. Maximum of 25 item(s) in the list.

Required: No

NextToken

The `nextToken` string returned on a previous page that you use to get the next page of results in a paginated response.

Type: String

Required: No

Response Syntax

```
{
    "ComplianceByConfigRules": [
        {
            "Compliance": {
                "ComplianceContributorCount": {
                    "CapExceeded": boolean,
                    "CappedCount": number
                },
                "ComplianceType": "string"
            },
            "ConfigRuleName": "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.

ComplianceByConfigRules

Indicates whether each of the specified AWS Config rules is compliant.

Type: array of ComplianceByConfigRule (p. 52) objects

NextToken

The string that you use in a subsequent request to get the next page of results in a paginated response.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 79).

InvalidParameterValueException

One or more of the specified parameters are invalid. Verify that your parameters are valid and try again.

HTTP Status Code: 400

NoSuchConfigRuleException

One or more AWS Config rules in the request are invalid. Verify that the rule names are correct and try again.

HTTP Status Code: 400

DescribeComplianceByResource

Indicates whether the specified AWS resources are compliant. If a resource is noncompliant, this action returns the number of AWS Config rules that the resource does not comply with.

A resource is compliant if it complies with all the AWS Config rules that evaluate it. It is noncompliant if it does not comply with one or more of these rules.

If AWS Config has no current evaluation results for the resource, it returns `INSUFFICIENT_DATA`. This result might indicate one of the following conditions about the rules that evaluate the resource:

- AWS Config has never invoked an evaluation for the rule. To check whether it has, use the `DescribeConfigRuleEvaluationStatus` action to get the `LastSuccessfulInvocationTime` and `LastFailedInvocationTime`.
- The rule's AWS Lambda function is failing to send evaluation results to AWS Config. Verify that the role that you assigned to your configuration recorder includes the `config:PutEvaluations` permission. If the rule is a customer managed rule, verify that the AWS Lambda execution role includes the `config:PutEvaluations` permission.
- The rule's AWS Lambda function has returned `NOT_APPLICABLE` for all evaluation results. This can occur if the resources were deleted or removed from the rule's scope.

Request Syntax

```
{
    "ComplianceTypes": [
        "string"
    ],
    "Limit": number,
    "NextToken": "string",
    "ResourceId": "string",
    "ResourceType": "string"
}
```

Request Parameters

For information about the common parameters that all actions use, see Common Parameters (p. 75).

The request requires the following data in JSON format.

ComplianceTypes

Filters the results by compliance.

The allowed values are `COMPLIANT`, `NON_COMPLIANT`, and `INSUFFICIENT_DATA`.

Type: array of Strings

Length constraints: Minimum of 0 item(s) in the list. Maximum of 3 item(s) in the list.

Required: No

Limit

The maximum number of evaluation results returned on each page. The default is 10. You cannot specify a limit greater than 100. If you specify 0, AWS Config uses the default.

Type: Number

Valid range: Minimum value of 0. Maximum value of 100.

Required: No

NextToken

The `nextToken` string returned on a previous page that you use to get the next page of results in a paginated response.

Type: String

Required: No

ResourceId

The ID of the AWS resource for which you want compliance information. You can specify only one resource ID. If you specify a resource ID, you must also specify a type for `ResourceType`.

Type: String

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

Required: No

ResourceType

The types of AWS resources for which you want compliance information; for example, `AWS::EC2::Instance`. For this action, you can specify that the resource type is an AWS account by specifying `AWS::::Account`.

Type: String

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

Required: No

Response Syntax

```
{
    "ComplianceByResources": [
        {
            "Compliance": {
                "ComplianceContributorCount": {
                    "CapExceeded": boolean,
                    "CappedCount": number
                },
                "ComplianceType": "string"
            },
            "ResourceId": "string",
            "ResourceType": "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.

ComplianceByResources
Indicates whether the specified AWS resource complies with all of the AWS Config rules that evaluate it.

Type: array of ComplianceByResource (p. 53) objects

NextToken
The string that you use in a subsequent request to get the next page of results in a paginated response.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 79).

InvalidNextTokenException
The specified next token is invalid. Specify the `nextToken` string that was returned in the previous response to get the next page of results.

HTTP Status Code: 400

InvalidParameterValueException
One or more of the specified parameters are invalid. Verify that your parameters are valid and try again.

HTTP Status Code: 400

DescribeConfigRuleEvaluationStatus

Returns status information for each of your AWS managed Config rules. The status includes information such as the last time AWS Config invoked the rule, the last time AWS Config failed to invoke the rule, and the related error for the last failure.

Request Syntax

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

Request Parameters

For information about the common parameters that all actions use, see Common Parameters (p. 75).

The request requires the following data in JSON format.

ConfigRuleNames

The name of the AWS managed Config rules for which you want status information. If you do not specify any names, AWS Config returns status information for all AWS managed Config rules that you use.

Type: array of Strings

Length constraints: Minimum of 0 item(s) in the list. Maximum of 25 item(s) in the list.

Required: No

Response Syntax

```
{
    "ConfigRulesEvaluationStatus": [
        {
            "ConfigRuleArn": "string",
            "ConfigRuleId": "string",
            "ConfigRuleName": "string",
            "FirstActivatedTime": number,
            "FirstEvaluationStarted": boolean,
            "LastErrorCode": "string",
            "LastErrorMessage": "string",
            "LastFailedEvaluationTime": number,
            "LastFailedInvocationTime": number,
            "LastSuccessfulEvaluationTime": number,
            "LastSuccessfulInvocationTime": 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.

ConfigRulesEvaluationStatus
   Status information about your AWS managed Config rules.

   Type: array of ConfigRuleEvaluationStatus (p. 58) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 79).

NoSuchConfigRuleException
   One or more AWS Config rules in the request are invalid. Verify that the rule names are correct and try again.

   HTTP Status Code: 400

DescribeConfigRules

Returns details about your AWS Config rules.

Request Syntax

```
{
    "ConfigRuleNames": [
        "string"
    ],
    "NextToken": "string"
}
```

Request Parameters

For information about the common parameters that all actions use, see Common Parameters (p. 75).

The request requires the following data in JSON format.

ConfigRuleNames
> The names of the AWS Config rules for which you want details. If you do not specify any names, AWS Config returns details for all your rules.
>
> Type: array of Strings
>
> Length constraints: Minimum of 0 item(s) in the list. Maximum of 25 item(s) in the list.
>
> Required: No

NextToken
> The nextToken string returned on a previous page that you use to get the next page of results in a paginated response.
>
> Type: String
>
> Required: No

Response Syntax

```
{
    "ConfigRules": [
        {
            "ConfigRuleArn": "string",
            "ConfigRuleId": "string",
            "ConfigRuleName": "string",
            "ConfigRuleState": "string",
            "Description": "string",
            "InputParameters": "string",
            "MaximumExecutionFrequency": "string",
            "Scope": {
                "ComplianceResourceId": "string",
                "ComplianceResourceTypes": [
```

```
                "string"
            ],
            "TagKey": "string",
            "TagValue": "string"
        },
        "Source": {
            "Owner": "string",
            "SourceDetails": [
                {
                    "EventSource": "string",
                    "MessageType": "string"
                }
            ],
            "SourceIdentifier": "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.

ConfigRules
    The details about your AWS Config rules.

    Type: array of ConfigRule (p. 56) objects
NextToken
    The string that you use in a subsequent request to get the next page of results in a paginated response.

    Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 79).

NoSuchConfigRuleException
    One or more AWS Config rules in the request are invalid. Verify that the rule names are correct and try again.

    HTTP Status Code: 400

DescribeConfigurationRecorders

Returns the name of one or more specified configuration recorders. If the recorder name is not specified, this action returns the names of all the configuration recorders associated with the account.

Note
Currently, you can specify only one configuration recorder per account.

Request Syntax

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

Request Parameters

For information about the common parameters that all actions use, see Common Parameters (p. 75).

The request requires the following data in JSON format.

ConfigurationRecorderNames
A list of configuration recorder names.

Type: array of Strings

Required: No

Response Syntax

```
{
    "ConfigurationRecorders": [
        {
            "name": "string",
            "recordingGroup": {
                "allSupported": boolean,
                "includeGlobalResourceTypes": boolean,
                "resourceTypes": [
                    "string"
                ]
            },
            "roleARN": "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.

ConfigurationRecorders
A list that contains the descriptions of the specified configuration recorders.

Type: array of ConfigurationRecorder (p. 63) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 79).

NoSuchConfigurationRecorderException
You have specified a configuration recorder that does not exist.

HTTP Status Code: 400

DescribeConfigurationRecorderStatus

Returns the current status of the specified configuration recorder. If a configuration recorder is not specified, this action returns the status of all configuration recorder associated with the account.

> **Note**
> Currently, you can specify only one configuration recorder per account.

Request Syntax

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

Request Parameters

For information about the common parameters that all actions use, see .

The request requires the following data in JSON format.

ConfigurationRecorderNames
The name(s) of the configuration recorder. If the name is not specified, the action returns the current status of all the configuration recorders associated with the account.

Type: array of Strings

Required: No

Response Syntax

```
{
    "ConfigurationRecordersStatus": [
        {
            "lastErrorCode": "string",
            "lastErrorMessage": "string",
            "lastStartTime": number,
            "lastStatus": "string",
            "lastStatusChangeTime": number,
            "lastStopTime": number,
            "name": "string",
            "recording": 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.

ConfigurationRecordersStatus
A list that contains status of the specified recorders.

Type: array of ConfigurationRecorderStatus (p. 63) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 79).

NoSuchConfigurationRecorderException
You have specified a configuration recorder that does not exist.

HTTP Status Code: 400

DescribeDeliveryChannels

Returns details about the specified delivery channel. If a delivery channel is not specified, this action returns the details of all delivery channels associated with the account.

Note
Currently, you can specify only one delivery channel per account.

Request Syntax

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

Request Parameters

For information about the common parameters that all actions use, see Common Parameters (p. 75).

The request requires the following data in JSON format.

DeliveryChannelNames
    A list of delivery channel names.

    Type: array of Strings

    Required: No

Response Syntax

```
{
    "DeliveryChannels": [
        {
            "configSnapshotDeliveryProperties": {
                "deliveryFrequency": "string"
            },
            "name": "string",
            "s3BucketName": "string",
            "s3KeyPrefix": "string",
            "snsTopicARN": "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.

DeliveryChannels

A list that contains the descriptions of the specified delivery channel.

Type: array of DeliveryChannel (p. 64) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 79).

NoSuchDeliveryChannelException

You have specified a delivery channel that does not exist.

HTTP Status Code: 400

DescribeDeliveryChannelStatus

Returns the current status of the specified delivery channel. If a delivery channel is not specified, this action returns the current status of all delivery channels associated with the account.

Note
Currently, you can specify only one delivery channel per account.

Request Syntax

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

Request Parameters

For information about the common parameters that all actions use, see .

The request requires the following data in JSON format.

DeliveryChannelNames
A list of delivery channel names.

Type: array of Strings

Required: No

Response Syntax

```
{
    "DeliveryChannelsStatus": [
        {
            "configHistoryDeliveryInfo": {
                "lastAttemptTime": number,
                "lastErrorCode": "string",
                "lastErrorMessage": "string",
                "lastStatus": "string",
                "lastSuccessfulTime": number,
                "nextDeliveryTime": number
            },
            "configSnapshotDeliveryInfo": {
                "lastAttemptTime": number,
                "lastErrorCode": "string",
                "lastErrorMessage": "string",
                "lastStatus": "string",
                "lastSuccessfulTime": number,
                "nextDeliveryTime": number
            },
            "configStreamDeliveryInfo": {
```

```
                "lastErrorCode": "string",
                "lastErrorMessage": "string",
                "lastStatus": "string",
                "lastStatusChangeTime": number
            },
            "name": "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.

DeliveryChannelsStatus
    A list that contains the status of a specified delivery channel.

    Type: array of DeliveryChannelStatus (p. 65) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 79).

NoSuchDeliveryChannelException
    You have specified a delivery channel that does not exist.

    HTTP Status Code: 400

GetComplianceDetailsByConfigRule

Returns the evaluation results for the specified AWS Config rule. The results indicate which AWS resources were evaluated by the rule, when each resource was last evaluated, and whether each resource complies with the rule.

Request Syntax

```
{
    "ComplianceTypes": [
        "string"
    ],
    "ConfigRuleName": "string",
    "Limit": number,
    "NextToken": "string"
}
```

Request Parameters

For information about the common parameters that all actions use, see Common Parameters (p. 75).

The request requires the following data in JSON format.

ComplianceTypes
Filters the results by compliance.

The allowed values are COMPLIANT, NON_COMPLIANT, and NOT_APPLICABLE.

Type: array of Strings

Length constraints: Minimum of 0 item(s) in the list. Maximum of 3 item(s) in the list.

Required: No

ConfigRuleName
The name of the AWS Config rule for which you want compliance information.

Type: String

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

Required: Yes

Limit
The maximum number of evaluation results returned on each page. The default is 10. You cannot specify a limit greater than 100. If you specify 0, AWS Config uses the default.

Type: Number

Valid range: Minimum value of 0. Maximum value of 100.

Required: No

NextToken
The nextToken string returned on a previous page that you use to get the next page of results in a paginated response.

Type: String

Required: No

Response Syntax

```
{
    "EvaluationResults": [
        {
            "Annotation": "string",
            "ComplianceType": "string",
            "ConfigRuleInvokedTime": number,
            "EvaluationResultIdentifier": {
                "EvaluationResultQualifier": {
                    "ConfigRuleName": "string",
                    "ResourceId": "string",
                    "ResourceType": "string"
                },
                "OrderingTimestamp": number
            },
            "ResultRecordedTime": number,
            "ResultToken": "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.

EvaluationResults
Indicates whether the AWS resource complies with the specified AWS Config rule.

Type: array of EvaluationResult (p. 67) objects

NextToken
The string that you use in a subsequent request to get the next page of results in a paginated response.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 79).

InvalidNextTokenException
The specified next token is invalid. Specify the nextToken string that was returned in the previous response to get the next page of results.

HTTP Status Code: 400

InvalidParameterValueException
One or more of the specified parameters are invalid. Verify that your parameters are valid and try again.

HTTP Status Code: 400

NoSuchConfigRuleException

One or more AWS Config rules in the request are invalid. Verify that the rule names are correct and try again.

HTTP Status Code: 400

GetComplianceDetailsByResource

Returns the evaluation results for the specified AWS resource. The results indicate which AWS Config rules were used to evaluate the resource, when each rule was last used, and whether the resource complies with each rule.

Request Syntax

```
{
    "ComplianceTypes": [
        "string"
    ],
    "NextToken": "string",
    "ResourceId": "string",
    "ResourceType": "string"
}
```

Request Parameters

For information about the common parameters that all actions use, see Common Parameters (p. 75).

The request requires the following data in JSON format.

ComplianceTypes
Filters the results by compliance.

The allowed values are COMPLIANT, NON_COMPLIANT, and NOT_APPLICABLE.

Type: array of Strings

Length constraints: Minimum of 0 item(s) in the list. Maximum of 3 item(s) in the list.

Required: No

NextToken
The nextToken string returned on a previous page that you use to get the next page of results in a paginated response.

Type: String

Required: No

ResourceId
The ID of the AWS resource for which you want compliance information.

Type: String

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

Required: Yes

ResourceType
The type of the AWS resource for which you want compliance information.

Type: String

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

Required: Yes

Response Syntax

```
{
    "EvaluationResults": [
        {
            "Annotation": "string",
            "ComplianceType": "string",
            "ConfigRuleInvokedTime": number,
            "EvaluationResultIdentifier": {
                "EvaluationResultQualifier": {
                    "ConfigRuleName": "string",
                    "ResourceId": "string",
                    "ResourceType": "string"
                },
                "OrderingTimestamp": number
            },
            "ResultRecordedTime": number,
            "ResultToken": "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.

EvaluationResults

Indicates whether the specified AWS resource complies each AWS Config rule.

Type: array of EvaluationResult (p. 67) objects

NextToken

The string that you use in a subsequent request to get the next page of results in a paginated response.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 79).

InvalidParameterValueException

One or more of the specified parameters are invalid. Verify that your parameters are valid and try again.

HTTP Status Code: 400

GetComplianceSummaryByConfigRule

Returns the number of AWS Config rules that are compliant and noncompliant, up to a maximum of 25 for each.

Response Syntax

```
{
    "ComplianceSummary": {
        "ComplianceSummaryTimestamp": number,
        "CompliantResourceCount": {
            "CapExceeded": boolean,
            "CappedCount": number
        },
        "NonCompliantResourceCount": {
            "CapExceeded": boolean,
            "CappedCount": 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.

ComplianceSummary

The number of AWS Config rules that are compliant and the number that are noncompliant, up to a maximum of 25 for each.

Type: ComplianceSummary (p. 54) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 79).

GetComplianceSummaryByResourceType

Returns the number of resources that are compliant and the number that are noncompliant. You can specify one or more resource types to get these numbers for each resource type. The maximum number returned is 100.

Request Syntax

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

Request Parameters

For information about the common parameters that all actions use, see Common Parameters (p. 75).

The request requires the following data in JSON format.

ResourceTypes
Specify one or more resource types to get the number of resources that are compliant and the number that are noncompliant for each resource type.

For this request, you can specify an AWS resource type such as `AWS::EC2::Instance`, and you can specify that the resource type is an AWS account by specifying `AWS::::Account`.

Type: array of Strings

Length constraints: Minimum of 0 item(s) in the list. Maximum of 20 item(s) in the list.

Required: No

Response Syntax

```
{
    "ComplianceSummariesByResourceType": [
        {
            "ComplianceSummary": {
                "ComplianceSummaryTimestamp": number,
                "CompliantResourceCount": {
                    "CapExceeded": boolean,
                    "CappedCount": number
                },
                "NonCompliantResourceCount": {
                    "CapExceeded": boolean,
                    "CappedCount": number
                }
            },
            "ResourceType": "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.

ComplianceSummariesByResourceType
> The number of resources that are compliant and the number that are noncompliant. If one or more resource types were provided with the request, the numbers are returned for each resource type. The maximum number returned is 100.
>
> Type: array of ComplianceSummaryByResourceType (p. 55) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 79).

InvalidParameterValueException
> One or more of the specified parameters are invalid. Verify that your parameters are valid and try again.
>
> HTTP Status Code: 400

GetResourceConfigHistory

Returns a list of configuration items for the specified resource. The list contains details about each state of the resource during the specified time interval.

The response is paginated, and by default, AWS Config returns a limit of 10 configuration items per page. You can customize this number with the `limit` parameter. The response includes a `nextToken` string, and to get the next page of results, run the request again and enter this string for the `nextToken` parameter.

> **Note**
> Each call to the API is limited to span a duration of seven days. It is likely that the number of records returned is smaller than the specified `limit`. In such cases, you can make another call, using the `nextToken`.

Request Syntax

```
{
    "chronologicalOrder": "string",
    "earlierTime": number,
    "laterTime": number,
    "limit": number,
    "nextToken": "string",
    "resourceId": "string",
    "resourceType": "string"
}
```

Request Parameters

For information about the common parameters that all actions use, see Common Parameters (p. 75).

The request requires the following data in JSON format.

chronologicalOrder
The chronological order for configuration items listed. By default the results are listed in reverse chronological order.

Type: String

Valid Values: `Reverse | Forward`

Required: No

earlierTime
The time stamp that indicates an earlier time. If not specified, the action returns paginated results that contain configuration items that start from when the first configuration item was recorded.

Type: DateTime

Required: No

laterTime
The time stamp that indicates a later time. If not specified, current time is taken.

Type: DateTime

Required: No

limit

The maximum number of configuration items returned on each page. The default is 10. You cannot specify a limit greater than 100. If you specify 0, AWS Config uses the default.

Type: Number

Valid range: Minimum value of 0. Maximum value of 100.

Required: No

nextToken

The `nextToken` string returned on a previous page that you use to get the next page of results in a paginated response.

Type: String

Required: No

resourceId

The ID of the resource (for example., `sg-xxxxxx`).

Type: String

Required: Yes

resourceType

The resource type.

Type: String

Valid Values: `AWS::EC2::CustomerGateway` | `AWS::EC2::EIP` | `AWS::EC2::Host` | `AWS::EC2::Instance` | `AWS::EC2::InternetGateway` | `AWS::EC2::NetworkAcl` | `AWS::EC2::NetworkInterface` | `AWS::EC2::RouteTable` | `AWS::EC2::SecurityGroup` | `AWS::EC2::Subnet` | `AWS::CloudTrail::Trail` | `AWS::EC2::Volume` | `AWS::EC2::VPC` | `AWS::EC2::VPNConnection` | `AWS::EC2::VPNGateway` | `AWS::IAM::Group` | `AWS::IAM::Policy` | `AWS::IAM::Role` | `AWS::IAM::User`

Required: Yes

Response Syntax

```
{
    "configurationItems": [
        {
            "accountId": "string",
            "arn": "string",
            "availabilityZone": "string",
            "awsRegion": "string",
            "configuration": "string",
            "configurationItemCaptureTime": number,
            "configurationItemMD5Hash": "string",
            "configurationItemStatus": "string",
            "configurationStateId": "string",
            "relatedEvents": [
                "string"
            ],
            "relationships": [
                {
```

```
                    "relationshipName": "string",
                    "resourceId": "string",
                    "resourceName": "string",
                    "resourceType": "string"
                }
            ],
            "resourceCreationTime": number,
            "resourceId": "string",
            "resourceName": "string",
            "resourceType": "string",
            "tags":
                {
                    "string" :
                        "string"
                },
            "version": "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.

configurationItems
  A list that contains the configuration history of one or more resources.

  Type: array of ConfigurationItem (p. 60) objects

nextToken
  The string that you use in a subsequent request to get the next page of results in a paginated response.

  Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 79).

InvalidLimitException
  The specified limit is outside the allowable range.

  HTTP Status Code: 400

InvalidNextTokenException
  The specified next token is invalid. Specify the `nextToken` string that was returned in the previous response to get the next page of results.

  HTTP Status Code: 400

InvalidTimeRangeException
  The specified time range is not valid. The earlier time is not chronologically before the later time.

  HTTP Status Code: 400

NoAvailableConfigurationRecorderException

There are no configuration recorders available to provide the role needed to describe your resources. Create a configuration recorder.

HTTP Status Code: 400

ResourceNotDiscoveredException

You have specified a resource that is either unknown or has not been discovered.

HTTP Status Code: 400

ValidationException

The requested action is not valid.

HTTP Status Code: 400

ListDiscoveredResources

Accepts a resource type and returns a list of resource identifiers for the resources of that type. A resource identifier includes the resource type, ID, and (if available) the custom resource name. The results consist of resources that AWS Config has discovered, including those that AWS Config is not currently recording. You can narrow the results to include only resources that have specific resource IDs or a resource name.

> **Note**
> You can specify either resource IDs or a resource name but not both in the same request.

The response is paginated, and by default AWS Config lists 100 resource identifiers on each page. You can customize this number with the `limit` parameter. The response includes a `nextToken` string, and to get the next page of results, run the request again and enter this string for the `nextToken` parameter.

Request Syntax

```
{
    "includeDeletedResources": boolean,
    "limit": number,
    "nextToken": "string",
    "resourceIds": [
        "string"
    ],
    "resourceName": "string",
    "resourceType": "string"
}
```

Request Parameters

For information about the common parameters that all actions use, see Common Parameters (p. 75).

The request requires the following data in JSON format.

includeDeletedResources
> Specifies whether AWS Config includes deleted resources in the results. By default, deleted resources are not included.
>
> Type: Boolean
>
> Required: No

limit
> The maximum number of resource identifiers returned on each page. The default is 100. You cannot specify a limit greater than 100. If you specify 0, AWS Config uses the default.
>
> Type: Number
>
> Valid range: Minimum value of 0. Maximum value of 100.
>
> Required: No

nextToken
> The `nextToken` string returned on a previous page that you use to get the next page of results in a paginated response.
>
> Type: String

Required: No

resourceIds

The IDs of only those resources that you want AWS Config to list in the response. If you do not specify this parameter, AWS Config lists all resources of the specified type that it has discovered.

Type: array of Strings

Required: No

resourceName

The custom name of only those resources that you want AWS Config to list in the response. If you do not specify this parameter, AWS Config lists all resources of the specified type that it has discovered.

Type: String

Required: No

resourceType

The type of resources that you want AWS Config to list in the response.

Type: String

Valid Values: `AWS::EC2::CustomerGateway | AWS::EC2::EIP | AWS::EC2::Host | AWS::EC2::Instance | AWS::EC2::InternetGateway | AWS::EC2::NetworkAcl | AWS::EC2::NetworkInterface | AWS::EC2::RouteTable | AWS::EC2::SecurityGroup | AWS::EC2::Subnet | AWS::CloudTrail::Trail | AWS::EC2::Volume | AWS::EC2::VPC | AWS::EC2::VPNConnection | AWS::EC2::VPNGateway | AWS::IAM::Group | AWS::IAM::Policy | AWS::IAM::Role | AWS::IAM::User`

Required: Yes

Response Syntax

```
{
    "nextToken": "string",
    "resourceIdentifiers": [
        {
            "resourceDeletionTime": number,
            "resourceId": "string",
            "resourceName": "string",
            "resourceType": "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

The string that you use in a subsequent request to get the next page of results in a paginated response.

Type: String

resourceIdentifiers

The details that identify a resource that is discovered by AWS Config, including the resource type, ID, and (if available) the custom resource name.

Type: array of ResourceIdentifier (p. 71) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 79).

InvalidLimitException

The specified limit is outside the allowable range.

HTTP Status Code: 400

InvalidNextTokenException

The specified next token is invalid. Specify the `nextToken` string that was returned in the previous response to get the next page of results.

HTTP Status Code: 400

NoAvailableConfigurationRecorderException

There are no configuration recorders available to provide the role needed to describe your resources. Create a configuration recorder.

HTTP Status Code: 400

ValidationException

The requested action is not valid.

HTTP Status Code: 400

PutConfigRule

Adds or updates an AWS Config rule for evaluating whether your AWS resources comply with your desired configurations.

You can use this action for customer managed Config rules and AWS managed Config rules. A customer managed Config rule is a custom rule that you develop and maintain. An AWS managed Config rule is a customizable, predefined rule that is provided by AWS Config.

If you are adding a new customer managed Config rule, you must first create the AWS Lambda function that the rule invokes to evaluate your resources. When you use the `PutConfigRule` action to add the rule to AWS Config, you must specify the Amazon Resource Name (ARN) that AWS Lambda assigns to the function. Specify the ARN for the `SourceIdentifier` key. This key is part of the `Source` object, which is part of the `ConfigRule` object.

If you are adding a new AWS managed Config rule, specify the rule's identifier for the `SourceIdentifier` key. To reference AWS managed Config rule identifiers, see Using AWS Managed Config Rules.

For any new rule that you add, specify the `ConfigRuleName` in the `ConfigRule` object. Do not specify the `ConfigRuleArn` or the `ConfigRuleId`. These values are generated by AWS Config for new rules.

If you are updating a rule that you have added previously, specify the rule's `ConfigRuleName`, `ConfigRuleId`, or `ConfigRuleArn` in the `ConfigRule` data type that you use in this request.

The maximum number of rules that AWS Config supports is 25.

For more information about developing and using AWS Config rules, see Evaluating AWS Resource Configurations with AWS Config in the *AWS Config Developer Guide*.

Request Syntax

```
{
    "ConfigRule": {
        "ConfigRuleArn": "string",
        "ConfigRuleId": "string",
        "ConfigRuleName": "string",
        "ConfigRuleState": "string",
        "Description": "string",
        "InputParameters": "string",
        "MaximumExecutionFrequency": "string",
        "Scope": {
            "ComplianceResourceId": "string",
            "ComplianceResourceTypes": [
                "string"
            ],
            "TagKey": "string",
            "TagValue": "string"
        },
        "Source": {
            "Owner": "string",
            "SourceDetails": [
                {
                    "EventSource": "string",
                    "MessageType": "string"
                }
```

```
            ],
            "SourceIdentifier": "string"
        }
    }
}
```

Request Parameters

For information about the common parameters that all actions use, see Common Parameters (p. 75).

The request requires the following data in JSON format.

ConfigRule

An AWS Lambda function that evaluates configuration items to assess whether your AWS resources comply with your desired configurations. This function can run when AWS Config detects a configuration change to an AWS resource, or when it delivers a configuration snapshot of the resources in the account.

For more information about developing and using AWS Config rules, see Evaluating AWS Resource Configurations with AWS Config in the *AWS Config Developer Guide*.

Type: ConfigRule (p. 56) 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 (p. 79).

InsufficientPermissionsException

Indicates one of the following errors:

- The rule cannot be created because the IAM role assigned to AWS Config lacks permissions to perform the config:Put* action.
- The AWS Lambda function cannot be invoked. Check the function ARN, and check the function's permissions.

HTTP Status Code: 400

InvalidParameterValueException

One or more of the specified parameters are invalid. Verify that your parameters are valid and try again.

HTTP Status Code: 400

MaxNumberOfConfigRulesExceededException

Failed to add the AWS Config rule because the account already contains the maximum number of 25 rules. Consider deleting any deactivated rules before adding new rules.

HTTP Status Code: 400

ResourceInUseException

The rule is currently being deleted. Wait for a while and try again.

HTTP Status Code: 400

PutConfigurationRecorder

Creates a new configuration recorder to record the selected resource configurations.

You can use this action to change the role `roleARN` and/or the `recordingGroup` of an existing recorder. To change the role, call the action on the existing configuration recorder and specify a role.

> **Note**
> Currently, you can specify only one configuration recorder per account.
> If `ConfigurationRecorder` does not have the **recordingGroup** parameter specified, the default is to record all supported resource types.

Request Syntax

```
{
    "ConfigurationRecorder": {
        "name": "string",
        "recordingGroup": {
            "allSupported": boolean,
            "includeGlobalResourceTypes": boolean,
            "resourceTypes": [
                "string"
            ]
        },
        "roleARN": "string"
    }
}
```

Request Parameters

For information about the common parameters that all actions use, see Common Parameters (p. 75).

The request requires the following data in JSON format.

ConfigurationRecorder
    The configuration recorder object that records each configuration change made to the resources.

    Type: ConfigurationRecorder (p. 63) 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 (p. 79).

InvalidConfigurationRecorderNameException
    You have provided a configuration recorder name that is not valid.

    HTTP Status Code: 400

InvalidRecordingGroupException

AWS Config throws an exception if the recording group does not contain a valid list of resource types. Invalid values could also be incorrectly formatted.

HTTP Status Code: 400

InvalidRoleException

You have provided a null or empty role ARN.

HTTP Status Code: 400

MaxNumberOfConfigurationRecordersExceededException

You have reached the limit on the number of recorders you can create.

HTTP Status Code: 400

PutDeliveryChannel

Creates a new delivery channel object to deliver the configuration information to an Amazon S3 bucket, and to an Amazon SNS topic.

You can use this action to change the Amazon S3 bucket or an Amazon SNS topic of the existing delivery channel. To change the Amazon S3 bucket or an Amazon SNS topic, call this action and specify the changed values for the S3 bucket and the SNS topic. If you specify a different value for either the S3 bucket or the SNS topic, this action will keep the existing value for the parameter that is not changed.

Note
Currently, you can specify only one delivery channel per account.

Request Syntax

```
{
    "DeliveryChannel": {
        "configSnapshotDeliveryProperties": {
            "deliveryFrequency": "string"
        },
        "name": "string",
        "s3BucketName": "string",
        "s3KeyPrefix": "string",
        "snsTopicARN": "string"
    }
}
```

Request Parameters

For information about the common parameters that all actions use, see Common Parameters (p. 75).

The request requires the following data in JSON format.

DeliveryChannel
The configuration delivery channel object that delivers the configuration information to an Amazon S3 bucket, and to an Amazon SNS topic.

Type: DeliveryChannel (p. 64) 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 (p. 79).

InsufficientDeliveryPolicyException
Your Amazon S3 bucket policy does not permit AWS Config to write to it.

HTTP Status Code: 400

InvalidDeliveryChannelNameException
The specified delivery channel name is not valid.

HTTP Status Code: 400

InvalidS3KeyPrefixException
The specified Amazon S3 key prefix is not valid.

HTTP Status Code: 400

InvalidSNSTopicARNException
The specified Amazon SNS topic does not exist.

HTTP Status Code: 400

MaxNumberOfDeliveryChannelsExceededException
You have reached the limit on the number of delivery channels you can create.

HTTP Status Code: 400

NoAvailableConfigurationRecorderException
There are no configuration recorders available to provide the role needed to describe your resources. Create a configuration recorder.

HTTP Status Code: 400

NoSuchBucketException
The specified Amazon S3 bucket does not exist.

HTTP Status Code: 400

PutEvaluations

Used by an AWS Lambda function to deliver evaluation results to AWS Config. This action is required in every AWS Lambda function that is invoked by an AWS Config rule.

Request Syntax

```
{
    "Evaluations": [
        {
            "Annotation": "string",
            "ComplianceResourceId": "string",
            "ComplianceResourceType": "string",
            "ComplianceType": "string",
            "OrderingTimestamp": number
        }
    ],
    "ResultToken": "string"
}
```

Request Parameters

For information about the common parameters that all actions use, see Common Parameters (p. 75).

The request requires the following data in JSON format.

Evaluations

The assessments that the AWS Lambda function performs. Each evaluation identifies an AWS resource and indicates whether it complies with the AWS Config rule that invokes the AWS Lambda function.

Type: array of Evaluation (p. 66) objects

Length constraints: Minimum of 0 item(s) in the list. Maximum of 100 item(s) in the list.

Required: No

ResultToken

An encrypted token that associates an evaluation with an AWS Config rule. Identifies the rule and the event that triggered the evaluation

Type: String

Required: Yes

Response Syntax

```
{
    "FailedEvaluations": [
        {
            "Annotation": "string",
            "ComplianceResourceId": "string",
```

```
            "ComplianceResourceType": "string",
            "ComplianceType": "string",
            "OrderingTimestamp": 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.

FailedEvaluations
> Requests that failed because of a client or server error.
>
> Type: array of Evaluation (p. 66) objects
>
> Length constraints: Minimum of 0 item(s) in the list. Maximum of 100 item(s) in the list.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 79).

InvalidParameterValueException
> One or more of the specified parameters are invalid. Verify that your parameters are valid and try again.
>
> HTTP Status Code: 400

InvalidResultTokenException
> The result token is invalid.
>
> HTTP Status Code: 400

NoSuchConfigRuleException
> One or more AWS Config rules in the request are invalid. Verify that the rule names are correct and try again.
>
> HTTP Status Code: 400

StartConfigurationRecorder

Starts recording configurations of the AWS resources you have selected to record in your AWS account.

You must have created at least one delivery channel to successfully start the configuration recorder.

Request Syntax

```
{
    "ConfigurationRecorderName": "string"
}
```

Request Parameters

For information about the common parameters that all actions use, see Common Parameters (p. 75).

The request requires the following data in JSON format.

ConfigurationRecorderName
The name of the recorder object that records each configuration change made to the resources.

Type: String

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

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. 79).

NoAvailableDeliveryChannelException
There is no delivery channel available to record configurations.

HTTP Status Code: 400

NoSuchConfigurationRecorderException
You have specified a configuration recorder that does not exist.

HTTP Status Code: 400

StopConfigurationRecorder

Stops recording configurations of the AWS resources you have selected to record in your AWS account.

Request Syntax

```
{
    "ConfigurationRecorderName": "string"
}
```

Request Parameters

For information about the common parameters that all actions use, see Common Parameters (p. 75).

The request requires the following data in JSON format.

ConfigurationRecorderName
> The name of the recorder object that records each configuration change made to the resources.
>
> Type: String
>
> Length constraints: Minimum length of 1. Maximum length of 256.
>
> 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. 79).

NoSuchConfigurationRecorderException
> You have specified a configuration recorder that does not exist.
>
> HTTP Status Code: 400

Data Types

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

> **Note**
> The order of each element in the response is not guaranteed. Applications should not assume a particular order.

The following data types are supported:

- Source (p. 73)
- SourceDetail (p. 73)

Compliance

Description

Indicates whether an AWS resource or AWS Config rule is compliant and provides the number of contributors that affect the compliance.

Contents

ComplianceContributorCount
>The number of AWS resources or AWS Config rules that cause a result of `NON_COMPLIANT`, up to a maximum number.
>
>Type: ComplianceContributorCount (p. 53) object
>
>Required: No

ComplianceType
>Indicates whether an AWS resource or AWS Config rule is compliant.
>
>A resource is compliant if it complies with all of the AWS Config rules that evaluate it, and it is noncompliant if it does not comply with one or more of these rules.
>
>A rule is compliant if all of the resources that the rule evaluates comply with it, and it is noncompliant if any of these resources do not comply.
>
>AWS Config returns the `INSUFFICIENT_DATA` value when no evaluation results are available for the AWS resource or Config rule.
>
>For the `Compliance` data type, AWS Config supports only `COMPLIANT`, `NON_COMPLIANT`, and `INSUFFICIENT_DATA` values. AWS Config does not support the `NOT_APPLICABLE` value for the `Compliance` data type.
>
>Type: String
>
>Valid Values: `COMPLIANT | NON_COMPLIANT | NOT_APPLICABLE | INSUFFICIENT_DATA`
>
>Required: No

ComplianceByConfigRule

Description

Indicates whether an AWS Config rule is compliant. A rule is compliant if all of the resources that the rule evaluated comply with it, and it is noncompliant if any of these resources do not comply.

Contents

Compliance
>Indicates whether the AWS Config rule is compliant.

Type: Compliance (p. 52) object

Required: No

ConfigRuleName
The name of the AWS Config rule.

Type: String

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

Required: No

ComplianceByResource

Description

Indicates whether an AWS resource that is evaluated according to one or more AWS Config rules is compliant. A resource is compliant if it complies with all of the rules that evaluate it, and it is noncompliant if it does not comply with one or more of these rules.

Contents

Compliance
Indicates whether the AWS resource complies with all of the AWS Config rules that evaluated it.

Type: Compliance (p. 52) object

Required: No

ResourceId
The ID of the AWS resource that was evaluated.

Type: String

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

Required: No

ResourceType
The type of the AWS resource that was evaluated.

Type: String

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

Required: No

ComplianceContributorCount

Description

The number of AWS resources or AWS Config rules responsible for the current compliance of the item, up to a maximum number.

Contents

CapExceeded

Indicates whether the maximum count is reached.

Type: Boolean

Required: No

CappedCount

The number of AWS resources or AWS Config rules responsible for the current compliance of the item.

Type: Number

Required: No

ComplianceSummary

Description

The number of AWS Config rules or AWS resources that are compliant and noncompliant, up to a maximum.

Contents

ComplianceSummaryTimestamp

The time that AWS Config created the compliance summary.

Type: DateTime

Required: No

CompliantResourceCount

The number of AWS Config rules or AWS resources that are compliant, up to a maximum of 25 for rules and 100 for resources.

Type: ComplianceContributorCount (p. 53) object

Required: No

NonCompliantResourceCount

The number of AWS Config rules or AWS resources that are noncompliant, up to a maximum of 25 for rules and 100 for resources.

Type: ComplianceContributorCount (p. 53) object

Required: No

ComplianceSummaryByResourceType

Description

The number of AWS resources of a specific type that are compliant or noncompliant, up to a maximum of 100 for each compliance.

Contents

ComplianceSummary
The number of AWS resources that are compliant or noncompliant, up to a maximum of 100 for each compliance.

Type: ComplianceSummary (p. 54) object

Required: No

ResourceType
The type of AWS resource.

Type: String

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

Required: No

ConfigExportDeliveryInfo

Description

A list that contains the status of the delivery of either the snapshot or the configuration history to the specified Amazon S3 bucket.

Contents

lastAttemptTime
The time of the last attempted delivery.

Type: DateTime

Required: No

lastErrorCode
The error code from the last attempted delivery.

Type: String

Required: No

lastErrorMessage
The error message from the last attempted delivery.

Type: String

Required: No

lastStatus
Status of the last attempted delivery.

Type: String

Valid Values: `Success | Failure | Not_Applicable`

Required: No

lastSuccessfulTime
The time of the last successful delivery.

Type: DateTime

Required: No

nextDeliveryTime
The time that the next delivery occurs.

Type: DateTime

Required: No

ConfigRule

Description

An AWS Lambda function that evaluates configuration items to assess whether your AWS resources comply with your desired configurations. This function can run when AWS Config detects a configuration change to an AWS resource, or when it delivers a configuration snapshot of the resources in the account.

For more information about developing and using AWS Config rules, see Evaluating AWS Resource Configurations with AWS Config in the *AWS Config Developer Guide*.

Contents

ConfigRuleArn
The Amazon Resource Name (ARN) of the AWS Config rule.

Type: String

Required: No

ConfigRuleId
The ID of the AWS Config rule.

Type: String

Required: No

ConfigRuleName
The name that you assign to the AWS Config rule. The name is required if you are adding a new rule.

Type: String

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

Required: No

ConfigRuleState

Indicates whether the AWS Config rule is active or currently being deleted by AWS Config.

AWS Config sets the state of a rule to `DELETING` temporarily after you use the `DeleteConfigRule` request to delete the rule. After AWS Config finishes deleting a rule, the rule and all of its evaluations are erased and no longer available.

You cannot add a rule to AWS Config that has the state set to `DELETING`. If you want to delete a rule, you must use the `DeleteConfigRule` request.

Type: String

Valid Values: `ACTIVE | DELETING`

Required: No

Description

The description that you provide for the AWS Config rule.

Type: String

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

Required: No

InputParameters

A string in JSON format that is passed to the AWS Config rule Lambda function.

Type: String

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

Required: No

MaximumExecutionFrequency

The maximum frequency at which the AWS Config rule runs evaluations.

If your rule is periodic, meaning it runs an evaluation when AWS Config delivers a configuration snapshot, then it cannot run evaluations more frequently than AWS Config delivers the snapshots. For periodic rules, set the value of the `MaximumExecutionFrequency` key to be equal to or greater than the value of the `deliveryFrequency` key, which is part of `ConfigSnapshotDeliveryProperties`. To update the frequency with which AWS Config delivers your snapshots, use the `PutDeliveryChannel` action.

Type: String

Valid Values: `One_Hour | Three_Hours | Six_Hours | Twelve_Hours | TwentyFour_Hours`

Required: No

Scope

Defines which resources can trigger an evaluation for the rule. The scope can include one or more resource types, a combination of one resource type and one resource ID, or a combination of a tag key and value. Specify a scope to constrain the resources that can trigger an evaluation for the rule. If you do not specify a scope, evaluations are triggered when any resource in the recording group changes.

Type: Scope (p. 72) object

Required: No

Source

Provides the rule owner (AWS or customer), the rule identifier, and the events that cause the function to evaluate your AWS resources.

Type: Source (p. 73) object

Required: Yes

ConfigRuleEvaluationStatus

Description

Status information for your AWS managed Config rules. The status includes information such as the last time the rule ran, the last time it failed, and the related error for the last failure.

This action does not return status information about customer managed Config rules.

Contents

ConfigRuleArn

The Amazon Resource Name (ARN) of the AWS Config rule.

Type: String

Required: No

ConfigRuleId

The ID of the AWS Config rule.

Type: String

Required: No

ConfigRuleName

The name of the AWS Config rule.

Type: String

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

Required: No

FirstActivatedTime

The time that you first activated the AWS Config rule.

Type: DateTime

Required: No

FirstEvaluationStarted

Indicates whether AWS Config has evaluated your resources against the rule at least once.

- `true` - AWS Config has evaluated your AWS resources against the rule at least once.
- `false` - AWS Config has not once finished evaluating your AWS resources against the rule.

Type: Boolean

Required: No

LastErrorCode

The error code that AWS Config returned when the rule last failed.

Type: String

Required: No

LastErrorMessage
The error message that AWS Config returned when the rule last failed.

Type: String

Required: No

LastFailedEvaluationTime
The time that AWS Config last failed to evaluate your AWS resources against the rule.

Type: DateTime

Required: No

LastFailedInvocationTime
The time that AWS Config last failed to invoke the AWS Config rule to evaluate your AWS resources.

Type: DateTime

Required: No

LastSuccessfulEvaluationTime
The time that AWS Config last successfully evaluated your AWS resources against the rule.

Type: DateTime

Required: No

LastSuccessfulInvocationTime
The time that AWS Config last successfully invoked the AWS Config rule to evaluate your AWS resources.

Type: DateTime

Required: No

ConfigSnapshotDeliveryProperties

Description

Options for how AWS Config delivers configuration snapshots to the Amazon S3 bucket in your delivery channel.

Contents

deliveryFrequency
The frequency with which a AWS Config recurringly delivers configuration snapshots.

Type: String

Valid Values: `One_Hour | Three_Hours | Six_Hours | Twelve_Hours | TwentyFour_Hours`

Required: No

ConfigStreamDeliveryInfo

Description

A list that contains the status of the delivery of the configuration stream notification to the Amazon SNS topic.

Contents

lastErrorCode
> The error code from the last attempted delivery.
>
> Type: String
>
> Required: No

lastErrorMessage
> The error message from the last attempted delivery.
>
> Type: String
>
> Required: No

lastStatus
> Status of the last attempted delivery.
>
> **Note** Providing an SNS topic on a DeliveryChannel for AWS Config is optional. If the SNS delivery is turned off, the last status will be **Not_Applicable**.
>
> Type: String
>
> Valid Values: `Success | Failure | Not_Applicable`
>
> Required: No

lastStatusChangeTime
> The time from the last status change.
>
> Type: DateTime
>
> Required: No

ConfigurationItem

Description

A list that contains detailed configurations of a specified resource.

> **Note**
> Currently, the list does not contain information about non-AWS components (for example, applications on your Amazon EC2 instances).

Contents

accountId
> The 12 digit AWS account ID associated with the resource.

Type: String

Required: No

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

Type: String

Required: No

availabilityZone
The Availability Zone associated with the resource.

Type: String

Required: No

awsRegion
The region where the resource resides.

Type: String

Required: No

configuration
The description of the resource configuration.

Type: String

Required: No

configurationItemCaptureTime
The time when the configuration recording was initiated.

Type: DateTime

Required: No

configurationItemMD5Hash
Unique MD5 hash that represents the configuration item's state.

You can use MD5 hash to compare the states of two or more configuration items that are associated with the same resource.

Type: String

Required: No

configurationItemStatus
The configuration item status.

Type: String

Valid Values: Ok | Failed | Discovered | Deleted

Required: No

configurationStateId
An identifier that indicates the ordering of the configuration items of a resource.

Type: String

Required: No

relatedEvents
A list of CloudTrail event IDs.

A populated field indicates that the current configuration was initiated by the events recorded in the CloudTrail log. For more information about CloudTrail, see What is AWS CloudTrail?.

An empty field indicates that the current configuration was not initiated by any event.

Type: array of Strings

Required: No

relationships

A list of related AWS resources.

Type: array of Relationship (p. 70) objects

Required: No

resourceCreationTime

The time stamp when the resource was created.

Type: DateTime

Required: No

resourceId

The ID of the resource (for example., `sg-xxxxxx`).

Type: String

Required: No

resourceName

The custom name of the resource, if available.

Type: String

Required: No

resourceType

The type of AWS resource.

Type: String

Valid Values: `AWS::EC2::CustomerGateway` | `AWS::EC2::EIP` | `AWS::EC2::Host` | `AWS::EC2::Instance` | `AWS::EC2::InternetGateway` | `AWS::EC2::NetworkAcl` | `AWS::EC2::NetworkInterface` | `AWS::EC2::RouteTable` | `AWS::EC2::SecurityGroup` | `AWS::EC2::Subnet` | `AWS::CloudTrail::Trail` | `AWS::EC2::Volume` | `AWS::EC2::VPC` | `AWS::EC2::VPNConnection` | `AWS::EC2::VPNGateway` | `AWS::IAM::Group` | `AWS::IAM::Policy` | `AWS::IAM::Role` | `AWS::IAM::User`

Required: No

tags

A mapping of key value tags associated with the resource.

Type: String to String map

Required: No

version

The version number of the resource configuration.

Type: String

Required: No

ConfigurationRecorder

Description

An object that represents the recording of configuration changes of an AWS resource.

Contents

name
> The name of the recorder. By default, AWS Config automatically assigns the name "default" when creating the configuration recorder. You cannot change the assigned name.
>
> Type: String
>
> Length constraints: Minimum length of 1. Maximum length of 256.
>
> Required: No

recordingGroup
> Specifies the types of AWS resource for which AWS Config records configuration changes.
>
> Type: RecordingGroup (p. 69) object
>
> Required: No

roleARN
> Amazon Resource Name (ARN) of the IAM role used to describe the AWS resources associated with the account.
>
> Type: String
>
> Required: No

ConfigurationRecorderStatus

Description

The current status of the configuration recorder.

Contents

lastErrorCode
> The error code indicating that the recording failed.
>
> Type: String
>
> Required: No

lastErrorMessage
> The message indicating that the recording failed due to an error.
>
> Type: String
>
> Required: No

lastStartTime
    The time the recorder was last started.

    Type: DateTime

    Required: No

lastStatus
    The last (previous) status of the recorder.

    Type: String

    Valid Values: `Pending | Success | Failure`

    Required: No

lastStatusChangeTime
    The time when the status was last changed.

    Type: DateTime

    Required: No

lastStopTime
    The time the recorder was last stopped.

    Type: DateTime

    Required: No

name
    The name of the configuration recorder.

    Type: String

    Required: No

recording
    Specifies whether the recorder is currently recording or not.

    Type: Boolean

    Required: No

DeliveryChannel

Description

A logical container used for storing the configuration changes of an AWS resource.

Contents

configSnapshotDeliveryProperties
    Options for how AWS Config delivers configuration snapshots to the Amazon S3 bucket in your delivery channel.

    Type: ConfigSnapshotDeliveryProperties (p. 59) object

    Required: No

name
> The name of the delivery channel. By default, AWS Config automatically assigns the name "default" when creating the delivery channel. You cannot change the assigned name.
>
> Type: String
>
> Length constraints: Minimum length of 1. Maximum length of 256.
>
> Required: No

s3BucketName
> The name of the Amazon S3 bucket used to store configuration history for the delivery channel.
>
> Type: String
>
> Required: No

s3KeyPrefix
> The prefix for the specified Amazon S3 bucket.
>
> Type: String
>
> Required: No

snsTopicARN
> The Amazon Resource Name (ARN) of the SNS topic that AWS Config delivers notifications to.
>
> Type: String
>
> Required: No

DeliveryChannelStatus

Description

The status of a specified delivery channel.

Valid values: `Success | Failure`

Contents

configHistoryDeliveryInfo
> A list that contains the status of the delivery of the configuration history to the specified Amazon S3 bucket.
>
> Type: ConfigExportDeliveryInfo (p. 55) object
>
> Required: No

configSnapshotDeliveryInfo
> A list containing the status of the delivery of the snapshot to the specified Amazon S3 bucket.
>
> Type: ConfigExportDeliveryInfo (p. 55) object
>
> Required: No

configStreamDeliveryInfo
> A list containing the status of the delivery of the configuration stream notification to the specified Amazon SNS topic.

Type: ConfigStreamDeliveryInfo (p. 60) object

Required: No

name
The name of the delivery channel.

Type: String

Required: No

Evaluation

Description

Identifies an AWS resource and indicates whether it complies with the AWS Config rule that it was evaluated against.

Contents

Annotation
Supplementary information about how the evaluation determined the compliance.

Type: String

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

Required: No

ComplianceResourceId
The ID of the AWS resource that was evaluated.

Type: String

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

Required: Yes

ComplianceResourceType
The type of AWS resource that was evaluated.

Type: String

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

Required: Yes

ComplianceType
Indicates whether the AWS resource complies with the AWS Config rule that it was evaluated against.

For the `Evaluation` data type, AWS Config supports only the `COMPLIANT`, `NON_COMPLIANT`, and `NOT_APPLICABLE` values. AWS Config does not support the `INSUFFICIENT_DATA` value for this data type.

Similarly, AWS Config does not accept `INSUFFICIENT_DATA` as the value for `ComplianceType` from a `PutEvaluations` request. For example, an AWS Lambda function for a custom Config rule cannot pass an `INSUFFICIENT_DATA` value to AWS Config.

Type: String

Valid Values: `COMPLIANT` | `NON_COMPLIANT` | `NOT_APPLICABLE` | `INSUFFICIENT_DATA`

Required: Yes

OrderingTimestamp

The time of the event in AWS Config that triggered the evaluation. For event-based evaluations, the time indicates when AWS Config created the configuration item that triggered the evaluation. For periodic evaluations, the time indicates when AWS Config delivered the configuration snapshot that triggered the evaluation.

Type: DateTime

Required: Yes

EvaluationResult

Description

The details of an AWS Config evaluation. Provides the AWS resource that was evaluated, the compliance of the resource, related timestamps, and supplementary information.

Contents

Annotation

Supplementary information about how the evaluation determined the compliance.

Type: String

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

Required: No

ComplianceType

Indicates whether the AWS resource complies with the AWS Config rule that evaluated it.

For the `EvaluationResult` data type, AWS Config supports only the `COMPLIANT`, `NON_COMPLIANT`, and `NOT_APPLICABLE` values. AWS Config does not support the `INSUFFICIENT_DATA` value for the `EvaluationResult` data type.

Type: String

Valid Values: `COMPLIANT` | `NON_COMPLIANT` | `NOT_APPLICABLE` | `INSUFFICIENT_DATA`

Required: No

ConfigRuleInvokedTime

The time when the AWS Config rule evaluated the AWS resource.

Type: DateTime

Required: No

EvaluationResultIdentifier

Uniquely identifies the evaluation result.

Type: EvaluationResultIdentifier (p. 68) object

Required: No

ResultRecordedTime
> The time when AWS Config recorded the evaluation result.
>
> Type: DateTime
>
> Required: No

ResultToken
> An encrypted token that associates an evaluation with an AWS Config rule. The token identifies the rule, the AWS resource being evaluated, and the event that triggered the evaluation.
>
> Type: String
>
> Required: No

EvaluationResultIdentifier

Description

Uniquely identifies an evaluation result.

Contents

EvaluationResultQualifier
> Identifies an AWS Config rule used to evaluate an AWS resource, and provides the type and ID of the evaluated resource.
>
> Type: EvaluationResultQualifier (p. 68) object
>
> Required: No

OrderingTimestamp
> The time of the event that triggered the evaluation of your AWS resources. The time can indicate when AWS Config delivered a configuration item change notification, or it can indicate when AWS Config delivered the configuration snapshot, depending on which event triggered the evaluation.
>
> Type: DateTime
>
> Required: No

EvaluationResultQualifier

Description

Identifies an AWS Config rule that evaluated an AWS resource, and provides the type and ID of the resource that the rule evaluated.

Contents

ConfigRuleName
> The name of the AWS Config rule that was used in the evaluation.
>
> Type: String

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

Required: No

ResourceId
The ID of the evaluated AWS resource.

Type: String

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

Required: No

ResourceType
The type of AWS resource that was evaluated.

Type: String

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

Required: No

RecordingGroup

Description

Specifies the types of AWS resource for which AWS Config records configuration changes.

In the recording group, you specify whether all supported types or specific types of resources are recorded.

By default, AWS Config records configuration changes for all supported types of regional resources that AWS Config discovers in the region in which it is running. Regional resources are tied to a region and can be used only in that region. Examples of regional resources are EC2 instances and EBS volumes.

You can also have AWS Config record configuration changes for supported types of global resources (for example, IAM resources). Global resources are not tied to an individual region and can be used in all regions.

> **Important**
> The configuration details for any global resource are the same in all regions. If you customize AWS Config in multiple regions to record global resources, it will create multiple configuration items each time a global resource changes: one configuration item for each region. These configuration items will contain identical data. To prevent duplicate configuration items, you should consider customizing AWS Config in only one region to record global resources, unless you want the configuration items to be available in multiple regions.

If you don't want AWS Config to record all resources, you can specify which types of resources it will record with the `resourceTypes` parameter.

For a list of supported resource types, see Supported resource types.

For more information, see Selecting Which Resources AWS Config Records.

Contents

allSupported
Specifies whether AWS Config records configuration changes for every supported type of regional resource.

If you set this option to `true`, when AWS Config adds support for a new type of regional resource, it automatically starts recording resources of that type.

If you set this option to `true`, you cannot enumerate a list of `resourceTypes`.

Type: Boolean

Required: No

includeGlobalResourceTypes
Specifies whether AWS Config includes all supported types of global resources (for example, IAM resources) with the resources that it records.

Before you can set this option to `true`, you must set the `allSupported` option to `true`.

If you set this option to `true`, when AWS Config adds support for a new type of global resource, it automatically starts recording resources of that type.

The configuration details for any global resource are the same in all regions. To prevent duplicate configuration items, you should consider customizing AWS Config in only one region to record global resources.

Type: Boolean

Required: No

resourceTypes
A comma-separated list that specifies the types of AWS resources for which AWS Config records configuration changes (for example, `AWS::EC2::Instance` or `AWS::CloudTrail::Trail`).

Before you can set this option to `true`, you must set the `allSupported` option to `false`.

If you set this option to `true`, when AWS Config adds support for a new type of resource, it will not record resources of that type unless you manually add that type to your recording group.

For a list of valid `resourceTypes` values, see the **resourceType Value** column in Supported AWS Resource Types.

Type: array of Strings

Required: No

Relationship

Description

The relationship of the related resource to the main resource.

Contents

relationshipName
The type of relationship with the related resource.

Type: String

Required: No

resourceId
The ID of the related resource (for example, `sg-xxxxxx`).

Type: String

Required: No

resourceName
> The custom name of the related resource, if available.

Type: String

Required: No

resourceType
> The resource type of the related resource.

Type: String

Valid Values: `AWS::EC2::CustomerGateway | AWS::EC2::EIP | AWS::EC2::Host | AWS::EC2::Instance | AWS::EC2::InternetGateway | AWS::EC2::NetworkAcl | AWS::EC2::NetworkInterface | AWS::EC2::RouteTable | AWS::EC2::SecurityGroup | AWS::EC2::Subnet | AWS::CloudTrail::Trail | AWS::EC2::Volume | AWS::EC2::VPC | AWS::EC2::VPNConnection | AWS::EC2::VPNGateway | AWS::IAM::Group | AWS::IAM::Policy | AWS::IAM::Role | AWS::IAM::User`

Required: No

ResourceIdentifier

Description

The details that identify a resource that is discovered by AWS Config, including the resource type, ID, and (if available) the custom resource name.

Contents

resourceDeletionTime
> The time that the resource was deleted.

Type: DateTime

Required: No

resourceId
> The ID of the resource (for example., `sg-xxxxxx`).

Type: String

Required: No

resourceName
> The custom name of the resource (if available).

Type: String

Required: No

resourceType
> The type of resource.

Type: String

Valid Values: `AWS::EC2::CustomerGateway` | `AWS::EC2::EIP` | `AWS::EC2::Host` | `AWS::EC2::Instance` | `AWS::EC2::InternetGateway` | `AWS::EC2::NetworkAcl` | `AWS::EC2::NetworkInterface` | `AWS::EC2::RouteTable` | `AWS::EC2::SecurityGroup` | `AWS::EC2::Subnet` | `AWS::CloudTrail::Trail` | `AWS::EC2::Volume` | `AWS::EC2::VPC` | `AWS::EC2::VPNConnection` | `AWS::EC2::VPNGateway` | `AWS::IAM::Group` | `AWS::IAM::Policy` | `AWS::IAM::Role` | `AWS::IAM::User`

Required: No

Scope

Description

Defines which resources trigger an evaluation for an AWS Config rule. The scope can include one or more resource types, a combination of a tag key and value, or a combination of one resource type and one resource ID. Specify a scope to constrain which resources trigger an evaluation for a rule. Otherwise, evaluations for the rule are triggered when any resource in your recording group changes in configuration.

Contents

ComplianceResourceId
The IDs of the only AWS resource that you want to trigger an evaluation for the rule. If you specify a resource ID, you must specify one resource type for `ComplianceResourceTypes`.

Type: String

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

Required: No

ComplianceResourceTypes
The resource types of only those AWS resources that you want to trigger an evaluation for the rule. You can only specify one type if you also specify a resource ID for `ComplianceResourceId`.

Type: array of Strings

Length constraints: Minimum of 0 item(s) in the list. Maximum of 100 item(s) in the list.

Required: No

TagKey
The tag key that is applied to only those AWS resources that you want you want to trigger an evaluation for the rule.

Type: String

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

Required: No

TagValue
The tag value applied to only those AWS resources that you want to trigger an evaluation for the rule. If you specify a value for `TagValue`, you must also specify a value for `TagKey`.

Type: String

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

Required: No

Source

Description

Provides the AWS Config rule owner (AWS or customer), the rule identifier, and the events that trigger the evaluation of your AWS resources.

Contents

Owner

Indicates whether AWS or the customer owns and manages the AWS Config rule.

Type: String

Valid Values: CUSTOM_LAMBDA | AWS

Required: No

SourceDetails

Provides the source and type of the event that causes AWS Config to evaluate your AWS resources.

Type: array of SourceDetail (p. 73) objects

Length constraints: Minimum of 0 item(s) in the list. Maximum of 25 item(s) in the list.

Required: No

SourceIdentifier

For AWS managed Config rules, a pre-defined identifier from a list. To reference the list, see Using AWS Managed Config Rules.

For customer managed Config rules, the identifier is the Amazon Resource Name (ARN) of the rule's AWS Lambda function.

Type: String

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

Required: No

SourceDetail

Description

Provides the source and type of the event that triggers AWS Config to evaluate your AWS resources against a rule.

Contents

EventSource

The source of the event, such as an AWS service, that triggers AWS Config to evaluate your AWS resources.

Type: String

Valid Values: `aws.config`

Required: No

MessageType

The type of SNS message that triggers AWS Config to run an evaluation. For evaluations that are initiated when AWS Config delivers a configuration item change notification, you must use `ConfigurationItemChangeNotification`. For evaluations that are initiated when AWS Config delivers a configuration snapshot, you must use `ConfigurationSnapshotDeliveryCompleted`.

Type: String

Valid Values: `ConfigurationItemChangeNotification | ConfigurationSnapshotDeliveryCompleted`

Required: No

Common Parameters

This section lists the request parameters that all actions use. Any action-specific parameters are listed in the topic for the action.

Action
The action to be performed.

Default: None

Type: string

Required: Yes

AuthParams
The parameters that are required to authenticate a Conditional request. Contains:

- AWSAccessKeyID
- SignatureVersion
- Timestamp
- Signature

Default: None

Required: Conditional

AWSAccessKeyId
The access key ID that corresponds to the secret access key that you used to sign the request.

Default: None

Type: string

Required: Yes

Expires
The date and time when the request signature expires, expressed in the format YYYY-MM-DDThh:mm:ssZ, as specified in the ISO 8601 standard.

Condition: Requests must include either *Timestamp* or *Expires*, but not both.

Default: None

Type: string

Required: Conditional

SecurityToken

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**.

Default: None

Type: string

Required: No

Signature

The digital signature that you created for the request. For information about generating a signature, go to the service's developer documentation.

Default: None

Type: string

Required: Yes

SignatureMethod

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

Default: None

Type: string

Valid Values: `HmacSHA256 | HmacSHA1`

Required: Yes

SignatureVersion

The signature version you use to sign the request. Set this to the value that is recommended for your service.

Default: None

Type: string

Required: Yes

Timestamp

The date and time when the request was signed, expressed in the format YYYY-MM-DDThh:mm:ssZ, as specified in the ISO 8601 standard.

Condition: Requests must include either *Timestamp* or *Expires*, but not both.

Default: None

Type: string

Required: Conditional

Version

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

Default: None

Type: string

Required: Yes

Common Parameters for Signature V4 Signing

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