AWS Control Tower

API Reference

API Version 2018-05-10

AWS Control Tower: API Reference

Table of Contents

Welcome

These interfaces allow you to apply the AWS library of pre-defined *controls* to your organizational units, programmatically. In this context, controls are the same as AWS Control Tower guardrails.

To call these APIs, you'll need to know:

- the `ControlARN` for the control--that is, the guardrail--you are targeting,
- and the ARN associated with the target organizational unit (OU).

To get the `ControlARN` for your AWS Control Tower guardrail:

The `ControlARN` contains the control name which is specified in each guardrail. For a list of control names for *Strongly recommended* and *Elective* guardrails, see Resource identifiers for APIs and guardrails in the Automating tasks section of the AWS Control Tower User Guide. Remember that *Mandatory* guardrails cannot be added or removed.

> **Note**
> **ARN format:** `arn:aws:controltower:{REGION}::control/{CONTROL_NAME}`
> **Example:**
> `arn:aws:controltower:us-west-2::control/AWS-GR_AUTOSCALING_LAUNCH_CONFIG_PUBLIC_IP_DISABLED`

To get the ARN for an OU:

In the AWS Organizations console, you can find the ARN for the OU on the **Organizational unit details** page associated with that OU.

> **Note**
> **OU ARN format:**
> `arn:${Partition}:organizations::${MasterAccountId}:ou/o-${OrganizationId}/ou-${OrganizationalUnitId}`

Details and examples

- List of resource identifiers for APIs and guardrails
- Guardrail API examples (CLI)
- Enable controls with AWS CloudFormation
- Creating AWS Control Tower resources with AWS CloudFormation

To view the open source resource repository on GitHub, see aws-cloudformation/aws-cloudformation-resource-providers-controltower

Recording API Requests

AWS Control Tower supports AWS CloudTrail, a service that records AWS API calls for your AWS account and delivers log files to an Amazon S3 bucket. By using information collected by CloudTrail, you can determine which requests the AWS Control Tower service received, who made the request and when, and so on. For more about AWS Control Tower and its support for CloudTrail, see Logging AWS Control Tower Actions with AWS CloudTrail in the AWS Control Tower User Guide. To learn more about CloudTrail, including how to turn it on and find your log files, see the AWS CloudTrail User Guide.

This document was last published on March 29, 2023.

Actions

The following actions are supported:

DisableControl

This API call turns off a control. It starts an asynchronous operation that deletes AWS resources on the specified organizational unit and the accounts it contains. The resources will vary according to the control that you specify.

Request Syntax

```
POST /disable-control HTTP/1.1
Content-type: application/json

{
   "controlIdentifier": "string",
   "targetIdentifier": "string"
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

controlIdentifier (p. 3)

The ARN of the control. Only **Strongly recommended** and **Elective** controls are permitted, with the exception of the **Region deny** guardrail.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `^arn:aws[0-9a-zA-Z_\-:\/]+$`

Required: Yes

targetIdentifier (p. 3)

The ARN of the organizational unit.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `^arn:aws[0-9a-zA-Z_\-:\/]+$`

Required: Yes

Response Syntax

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

{
```

```
   "operationIdentifier": "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.

operationIdentifier (p. 3)

The ID of the asynchronous operation, which is used to track status. The operation is available for 90 days.

Type: String

Length Constraints: Fixed length of 36.

Pattern: `^[a-f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 19).

AccessDeniedException

User does not have sufficient access to perform this action.

HTTP Status Code: 403

ConflictException

Updating or deleting a resource can cause an inconsistent state.

HTTP Status Code: 409

InternalServerException

Unexpected error during processing of request.

HTTP Status Code: 500

ResourceNotFoundException

Request references a resource which does not exist.

HTTP Status Code: 404

ServiceQuotaExceededException

Request would cause a service quota to be exceeded. The limit is 10 concurrent operations.

HTTP Status Code: 402

ThrottlingException

Request was denied due to request throttling.

HTTP Status Code: 429

ValidationException

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

HTTP Status Code: 400

See Also

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

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

EnableControl

This API call activates a control. It starts an asynchronous operation that creates AWS resources on the specified organizational unit and the accounts it contains. The resources created will vary according to the control that you specify.

Request Syntax

```
POST /enable-control HTTP/1.1
Content-type: application/json

{
   "controlIdentifier": "string",
   "targetIdentifier": "string"
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

controlIdentifier (p. 6)

The ARN of the control. Only **Strongly recommended** and **Elective** controls are permitted, with the exception of the **Region deny** guardrail.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `^arn:aws[0-9a-zA-Z_\-:\/]+$`

Required: Yes

targetIdentifier (p. 6)

The ARN of the organizational unit.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `^arn:aws[0-9a-zA-Z_\-:\/]+$`

Required: Yes

Response Syntax

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

{
```

```
   "operationIdentifier": "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.

operationIdentifier (p. 6)

The ID of the asynchronous operation, which is used to track status. The operation is available for 90 days.

Type: String

Length Constraints: Fixed length of 36.

Pattern: `^[a-f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 19).

AccessDeniedException

User does not have sufficient access to perform this action.

HTTP Status Code: 403

ConflictException

Updating or deleting a resource can cause an inconsistent state.

HTTP Status Code: 409

InternalServerException

Unexpected error during processing of request.

HTTP Status Code: 500

ResourceNotFoundException

Request references a resource which does not exist.

HTTP Status Code: 404

ServiceQuotaExceededException

Request would cause a service quota to be exceeded. The limit is 10 concurrent operations.

HTTP Status Code: 402

ThrottlingException

Request was denied due to request throttling.

HTTP Status Code: 429

ValidationException

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

HTTP Status Code: 400

See Also

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

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetControlOperation

Returns the status of a particular `EnableControl` or `DisableControl` operation. Displays a message in case of error. Details for an operation are available for 90 days.

Request Syntax

```
POST /get-control-operation HTTP/1.1
Content-type: application/json

{
   "operationIdentifier": "string"
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

operationIdentifier (p. 9)

The ID of the asynchronous operation, which is used to track status. The operation is available for 90 days.

Type: String

Length Constraints: Fixed length of 36.

Pattern: `^[a-f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$`

Required: Yes

Response Syntax

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

{
   "controlOperation": {
      "endTime": number,
      "operationType": "string",
      "startTime": number,
      "status": "string",
      "statusMessage": "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.

controlOperation (p. 9)

Type: ControlOperation (p. 15) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 19).

AccessDeniedException

User does not have sufficient access to perform this action.

HTTP Status Code: 403

InternalServerException

Unexpected error during processing of request.

HTTP Status Code: 500

ResourceNotFoundException

Request references a resource which does not exist.

HTTP Status Code: 404

ThrottlingException

Request was denied due to request throttling.

HTTP Status Code: 429

ValidationException

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

HTTP Status Code: 400

See Also

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

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

ListEnabledControls

Lists the controls enabled by AWS Control Tower on the specified organizational unit and the accounts it contains.

Request Syntax

```
POST /list-enabled-controls HTTP/1.1
Content-type: application/json

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

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

maxResults (p. 11)

How many results to return per API call.

Type: Integer

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

Required: No

nextToken (p. 11)

The token to continue the list from a previous API call with the same parameters.

Type: String

Required: No

targetIdentifier (p. 11)

The ARN of the organizational unit.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `^arn:aws[0-9a-zA-Z_\-:\/]+$`

Required: Yes

Response Syntax

```
HTTP/1.1 200
```

```
Content-type: application/json

{
   "enabledControls": [
      {
         "controlIdentifier": "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.

enabledControls (p. 11)

Lists the controls enabled by AWS Control Tower on the specified organizational unit and the accounts it contains.

Type: Array of EnabledControlSummary (p. 16) objects

nextToken (p. 11)

Retrieves the next page of results. If the string is empty, the current response is the end of the results.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 19).

AccessDeniedException

User does not have sufficient access to perform this action.

HTTP Status Code: 403

InternalServerException

Unexpected error during processing of request.

HTTP Status Code: 500

ResourceNotFoundException

Request references a resource which does not exist.

HTTP Status Code: 404

ThrottlingException

Request was denied due to request throttling.

HTTP Status Code: 429

ValidationException

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

HTTP Status Code: 400

See Also

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

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Data Types

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

> **Note**
> The order of each element in a data type structure is not guaranteed. Applications should not assume a particular order.

The following data types are supported:

ControlOperation

An operation performed by the control.

Contents

endTime

The time that the operation finished.

Type: Timestamp

Required: No

operationType

One of `ENABLE_CONTROL` or `DISABLE_CONTROL`.

Type: String

Valid Values: `ENABLE_CONTROL | DISABLE_CONTROL`

Required: No

startTime

The time that the operation began.

Type: Timestamp

Required: No

status

One of `IN_PROGRESS`, `SUCEEDED`, or `FAILED`.

Type: String

Valid Values: `SUCCEEDED | FAILED | IN_PROGRESS`

Required: No

statusMessage

If the operation result is `FAILED`, this string contains a message explaining why the operation failed.

Type: String

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

EnabledControlSummary

A summary of enabled controls.

Contents

controlIdentifier

The ARN of the control. Only **Strongly recommended** and **Elective** controls are permitted, with the exception of the **Region deny** guardrail.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `^arn:aws[0-9a-zA-Z_\-:\/]+$`

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Common Parameters

The following list contains the parameters that all actions use for signing Signature Version 4 requests with a query string. Any action-specific parameters are listed in the topic for that action. For more information about Signature Version 4, see Signature Version 4 Signing Process in the *Amazon Web Services General Reference*.

Action

The action to be performed.

Type: string

Required: Yes

Version

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

Type: string

Required: Yes

X-Amz-Algorithm

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

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

Type: string

Valid Values: `AWS4-HMAC-SHA256`

Required: Conditional

X-Amz-Credential

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

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

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

Type: string

Required: Conditional

X-Amz-Date

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

Condition: X-Amz-Date is optional for all requests; it can be used to override the date used for signing requests. If the Date header is specified in the ISO 8601 basic format, X-Amz-Date is

not required. When X-Amz-Date is used, it always overrides the value of the Date header. For more information, see Handling Dates in Signature Version 4 in the *Amazon Web Services General Reference*.

Type: string

Required: Conditional

X-Amz-Security-Token

The temporary security token that was obtained through a call to AWS Security Token Service (AWS STS). For a list of services that support temporary security credentials from AWS Security Token Service, go to AWS Services That Work with IAM in the *IAM User Guide*.

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

Type: string

Required: Conditional

X-Amz-Signature

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

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

Type: string

Required: Conditional

X-Amz-SignedHeaders

Specifies all the HTTP headers that were included as part of the canonical request. For more information about specifying signed headers, see Task 1: Create a Canonical Request For Signature Version 4 in the *Amazon Web Services General Reference*.

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

Type: string

Required: Conditional

Common Errors

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

AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 400

IncompleteSignature

The request signature does not conform to AWS standards.

HTTP Status Code: 400

InternalFailure

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

HTTP Status Code: 500

InvalidAction

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

HTTP Status Code: 400

InvalidClientTokenId

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

HTTP Status Code: 403

InvalidParameterCombination

Parameters that must not be used together were used together.

HTTP Status Code: 400

InvalidParameterValue

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

HTTP Status Code: 400

InvalidQueryParameter

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

HTTP Status Code: 400

MalformedQueryString

The query string contains a syntax error.

HTTP Status Code: 404

MissingAction

The request is missing an action or a required parameter.

HTTP Status Code: 400

MissingAuthenticationToken

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

HTTP Status Code: 403

MissingParameter

A required parameter for the specified action is not supplied.

HTTP Status Code: 400

NotAuthorized

You do not have permission to perform this action.

HTTP Status Code: 400

OptInRequired

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

HTTP Status Code: 403

RequestExpired

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

HTTP Status Code: 400

ServiceUnavailable

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

HTTP Status Code: 503

ThrottlingException

The request was denied due to request throttling.

HTTP Status Code: 400

ValidationError

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

HTTP Status Code: 400