AWS Resource Groups

Resource Groups API Reference

AWS Resource Groups: Resource Groups API Reference

Table of Contents

Welcome

AWS Resource Groups lets you organize AWS resources such as Amazon Elastic Compute Cloud instances, Amazon Relational Database Service databases, and Amazon Simple Storage Service buckets into groups using criteria that you define as tags. A resource group is a collection of resources that match the resource types specified in a query, and share one or more tags or portions of tags. You can create a group of resources based on their roles in your cloud infrastructure, lifecycle stages, regions, application layers, or virtually any criteria. Resource Groups enable you to automate management tasks, such as those in AWS Systems Manager Automation documents, on tag-related resources in AWS Systems Manager. Groups of tagged resources also let you quickly view a custom console in AWS Systems Manager that shows AWS Config compliance and other monitoring data about member resources.

To create a resource group, build a resource query, and specify tags that identify the criteria that members of the group have in common. Tags are key-value pairs.

For more information about Resource Groups, see the AWS Resource Groups User Guide.

AWS Resource Groups uses a REST-compliant API that you can use to perform the following types of operations.

- Create, Read, Update, and Delete (CRUD) operations on resource groups and resource query entities
- Applying, editing, and removing tags from resource groups
- Resolving resource group member ARNs so they can be returned as search results
- Getting data about resources that are members of a group
- Searching AWS resources based on a resource query

This document was last published on June 6, 2022.

Actions

The following actions are supported:

CreateGroup

Creates a resource group with the specified name and description. You can optionally include either a resource query or a service configuration. For more information about constructing a resource query, see Build queries and groups in AWS Resource Groups in the *AWS Resource Groups User Guide*. For more information about service-linked groups and service configurations, see Service configurations for Resource Groups.

Minimum permissions

To run this command, you must have the following permissions:

- `resource-groups:CreateGroup`

Request Syntax

```
POST /groups HTTP/1.1
Content-type: application/json

{
   "Configuration": [
      {
         "Parameters": [
            {
               "Name": "string",
               "Values": [ "string" ]
            }
         ],
         "Type": "string"
      }
   ],
   "Description": "string",
   "Name": "string",
   "ResourceQuery": {
      "Query": "string",
      "Type": "string"
   },
   "Tags": {
      "string" : "string"
   }
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

Name (p. 3)

The name of the group, which is the identifier of the group in other operations. You can't change the name of a resource group after you create it. A resource group name can consist of letters, numbers, hyphens, periods, and underscores. The name cannot start with `AWS` or `aws`; these are reserved. A resource group name must be unique within each AWS Region in your AWS account.

Type: String

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

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

Required: Yes

Configuration (p. 3)

A configuration associates the resource group with an AWS service and specifies how the service can interact with the resources in the group. A configuration is an array of GroupConfigurationItem (p. 84) elements. For details about the syntax of service configurations, see Service configurations for Resource Groups.

Note
A resource group can contain either a `Configuration` or a `ResourceQuery`, but not both.

Type: Array of GroupConfigurationItem (p. 84) objects

Array Members: Maximum number of 2 items.

Required: No

Description (p. 3)

The description of the resource group. Descriptions can consist of letters, numbers, hyphens, underscores, periods, and spaces.

Type: String

Length Constraints: Maximum length of 512.

Pattern: `[\sa-zA-Z0-9_\.-]*`

Required: No

ResourceQuery (p. 3)

The resource query that determines which AWS resources are members of this group. For more information about resource queries, see Create a tag-based group in Resource Groups.

Note
A resource group can contain either a `ResourceQuery` or a `Configuration`, but not both.

Type: ResourceQuery (p. 94) object

Required: No

Tags (p. 3)

The tags to add to the group. A tag is key-value pair string.

Type: String to string map

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

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

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

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

Required: No

Response Syntax

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

{
   "Group": {
      "Description": "string",
      "GroupArn": "string",
      "Name": "string"
   },
   "GroupConfiguration": {
      "Configuration": [
         {
            "Parameters": [
               {
                  "Name": "string",
                  "Values": [ "string" ]
               }
            ],
            "Type": "string"
         }
      ],
      "FailureReason": "string",
      "ProposedConfiguration": [
         {
            "Parameters": [
               {
                  "Name": "string",
                  "Values": [ "string" ]
               }
            ],
            "Type": "string"
         }
      ],
      "Status": "string"
   },
   "ResourceQuery": {
      "Query": "string",
      "Type": "string"
   },
   "Tags": {
      "string" : "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.

Group (p. 5)

The description of the resource group.

Type: Group (p. 80) object

GroupConfiguration (p. 5)

The service configuration associated with the resource group. For details about the syntax of a service configuration, see Service configurations for Resource Groups.

Type: GroupConfiguration (p. 82) object

ResourceQuery (p. 5)

The resource query associated with the group. For more information about resource queries, see Create a tag-based group in Resource Groups.

Type: ResourceQuery (p. 94) object

Tags (p. 5)

The tags associated with the group.

Type: String to string map

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

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

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

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 111).

BadRequestException

The request includes one or more parameters that violate validation rules.

HTTP Status Code: 400

ForbiddenException

The caller isn't authorized to make the request. Check permissions.

HTTP Status Code: 403

InternalServerErrorException

An internal error occurred while processing the request. Try again later.

HTTP Status Code: 500

MethodNotAllowedException

The request uses an HTTP method that isn't allowed for the specified resource.

HTTP Status Code: 405

TooManyRequestsException

You've exceeded throttling limits by making too many requests in a period of time.

HTTP Status Code: 429

Examples

Example

The following example creates a resource group in the `us-west-2` Region of the calling account. The resource group has a query that specifies that any resources in the account that are tagged with the key

`Stage` and a value of `Test` are members of the group. The group itself (not its members) is tagged with a key named `Department` and a value of `Finance`.

Sample Request

```
POST /groups HTTP/1.1
Host: resource-groups.us-west-2.amazonaws.com
Accept-Encoding: identity
User-Agent: aws-cli/2.2.40 Python/3.8.8 Windows/10 exe/AMD64 prompt/off command/resource-
groups.create-group
X-Amz-Date: 20220113T175008Z
X-Amz-Security-Token: <SECURITY-TOKEN>
Authorization: AWS4-HMAC-SHA256 Credential=<ACCESS-KEY>/20220113/us-west-2/resource-groups/
aws4_request,SignedHeaders=host;x-amz-date;x-amz-security-token,Signature=<SIGV4-SIGNATURE>
Content-Length: 283

{
    "Description": "Resources created for the testing stage.",
    "Name": "QueryGroup",
    "ResourceQuery": {
        "Query": "{\"ResourceTypeFilters\":[\"AWS::AllSupported\"],\"TagFilters\":[{\"Key
\":\"Stage\",\"Values\":[\"Test\"]}]}",
         "Type": "TAG_FILTERS_1_0"
    },
    "Tags": {"Department": "Finance"}
}
```

```
HTTP/1.1 200 OK
Date: Thu, 13 Jan 2022 17:50:08 GMT
Content-Type: application/json
Content-Length: 384
x-amzn-RequestId: <VARIES>
x-amz-apigw-id: <VARIES>
X-Amzn-Trace-Id: Root=<VARIES>
Connection: keep-alive

{
    "Group":{
        "GroupArn":"arn:aws:resource-groups:us-west-2:123456789012:group/QueryGroup",
        "Name":"QueryGroup",
        "Description":"Resources created for the testing stage.",
        "OwnerId":"123456789012"
    },
    "ResourceQuery":{
        "Type":"TAG_FILTERS_1_0",
        "Query":"{\"ResourceTypeFilters\":[\"AWS::AllSupported\"],\"TagFilters\":[{\"Key\":
\"Stage\",\"Values\":[\"Test\"]}]}"
    },
    "Tags":{"Department":"Finance"}
}
```

Example

The following example creates a resource group with a configuration that makes the group serve as a capacity reservation pool by Amazon Elastic Compute Cloud. This group can contain only Amazon EC2 capacity reservations that you add by using the GroupResources (p. 33) operation.

Sample Request

```
POST /groups HTTP/1.1
```

```
Host: resource-groups.us-west-2.amazonaws.com
Accept-Encoding: identity
User-Agent: aws-cli/2.2.40 Python/3.8.8 Windows/10 exe/AMD64 prompt/off command/resource-
groups.create-group
X-Amz-Date: 20220113T180534Z
X-Amz-Security-Token: <SECURITY-TOKEN>
Authorization: AWS4-HMAC-SHA256 Credential=<ACCESS-KEY>/20220113/us-west-2/resource-
groups/aws4_request,SignedHeaders=host;x-amz-date;x-amz-security-token,Signature=<SIGV4-
SIGNATURE>Content-Length: 320
Content-Length: 320

{
    "Name": "CRPGroup",
    "Description": "Resource group for capacity reservations.",
    "Configuration": [
        {
            "Type": "AWS::EC2::CapacityReservationPool"
        },
        {
            "Type": "AWS::ResourceGroups::Generic",
            "Parameters": [
                {
                    "Name": "allowed-resource-types",
                    "Values": [
                        "AWS::EC2::CapacityReservation"
                    ]
                }
            ]
        }
    ],
    "Tags": {
        "Department": "Finance"
    }
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: Thu, 13 Jan 2022 18:05:34 GMT
Content-Type: application/json
Content-Length: 561
x-amzn-RequestId: <VARIES>
x-amz-apigw-id: <VARIES>
X-Amzn-Trace-Id: Root=<VARIES>
Connection: keep-alive

{
    "Group":{
        "GroupArn":"arn:aws:resource-groups:us-west-2:123456789012:group/CRPGroup",
        "Name":"CRPGroup",
        "Description":"Resource group for capacity reservations.",
        "OwnerId":"123456789012"
    },
    "Tags":{
        "Department":"Finance"
    },
    "GroupConfiguration":{
        "GroupParameters":[
            {
                "Name":"allowed-resource-types",
                "Values":[
                    "AWS::EC2::CapacityReservation"
                ]
            }
```

```
        ],
        "Configuration":[
            {
                "Type":"AWS::EC2::CapacityReservationPool"
            },
            {
                "Type":"AWS::ResourceGroups::Generic",
                "Parameters":[
                    {
                        "Name":"allowed-resource-types",
                        "Values":[
                            "AWS::EC2::CapacityReservation"
                        ]
                    }
                ]
            }
        ],
        "Status": "UPDATE_COMPLETE"
    }
}
```

Example

The following example creates a resource group that can contain only Amazon EC2 hosts. Each instance launched into the group is automatically configured to use any of the available core/socket based license configurations you have defined in AWS License Manager.

Sample Request

```
POST /groups HTTP/1.1
Host: resource-groups.us-west-2.amazonaws.com
Accept-Encoding: identity
User-Agent: aws-cli/2.2.40 Python/3.8.8 Windows/10 exe/AMD64 prompt/off command/resource-
groups.create-group
X-Amz-Date: 20220113T182832Z
X-Amz-Security-Token: <SECURITY-TOKEN>
Authorization: AWS4-HMAC-SHA256 Credential=<ACCESS-KEY>/20220113/us-west-2/resource-groups/
aws4_request,SignedHeaders=host;x-amz-date;x-amz-security-token,Signature=<SIGV4-SIGNATURE>
Content-Length: 362

{
    "Name": "HostManagementGroup",
    "Configuration": [
        {
            "Type": "AWS::EC2::HostManagement",
            "Parameters": [
                {
                    "Name": "any-host-based-license-configuration",
                    "Values": [
                        "true"
                    ]
                }
            ]
        },
        {
            "Type": "AWS::ResourceGroups::Generic",
            "Parameters": [
                {
                    "Name": "allowed-resource-types",
                    "Values": [
                        "AWS::EC2::Host"
                    ]
                },
```

```
                {
                    "Name": "deletion-protection",
                    "Values": [
                        "UNLESS_EMPTY"
                    ]
                }
            ]
        }
    ]
}
```

```
 role="response">HTTP/1.1 200 OK
Date: Thu, 13 Jan 2022 18:28:33 GMT
Content-Type: application/json
Content-Length: 881
x-amzn-RequestId: <VARIES>
x-amz-apigw-id: <VARIES>
X-Amzn-Trace-Id: Root=<VARIES>
Connection: keep-alive

{
    Group":{
        "GroupArn":"arn:aws:resource-groups:us-west-2:123456789012:group/
HostManagementGroup",
        "Name":"HostManagementGroup",
        "OwnerId":"123456789012"
    },
    "GroupConfiguration":{
        "GroupParameters":[
            {
                "Name":"allowed-resource-types",
                "Values":[
                    "AWS::EC2::Host"
                ]
            },
            {
                "Name":"deletion-protection",
                "Values":[
                    "UNLESS_EMPTY"
                ]
            }
        ],
        "ResourceTypeParameters":[
            {
                "ResourceType":"AWS::EC2::Host",
                "Name":"exclusive-membership",
                "Values":[
                    "ACROSS_SAME_SERVICE_LINK_TYPE"
                ]
            }
        ],
        "ServiceLinkParameters":[
            {
                "Name":"any-host-based-license-configuration",
                "Values":[
                    "true"
                ]
            }
        ],
        "Configuration":[
            {
                "Type":"AWS::EC2::HostManagement",
                "Parameters":[
                    {
```

```
                        "Name":"any-host-based-license-configuration",
                        "Values":[
                            "true"
                        ]
                    }
                ]
            },
            {
                "Type":"AWS::ResourceGroups::Generic",
                "Parameters":[
                    {
                        "Name":"allowed-resource-types",
                        "Values":[
                            "AWS::EC2::Host"
                        ]
                    },
                    {
                        "Name":"deletion-protection",
                        "Values":[
                            "UNLESS_EMPTY"
                        ]
                    }
                ]
            }
        ],
        "Status":"UPDATE_COMPLETE"
    }
}
```


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

DeleteGroup

Deletes the specified resource group. Deleting a resource group does not delete any resources that are members of the group; it only deletes the group structure.

Minimum permissions

To run this command, you must have the following permissions:

- `resource-groups:DeleteGroup`

Request Syntax

```
POST /delete-group HTTP/1.1
Content-type: application/json

{
   "Group": "string",
   "GroupName": "string"
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

Group (p. 12)

The name or the ARN of the resource group to delete.

Type: String

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

Pattern: `(arn:aws(-[a-z]+)*:resource-groups:[a-z]{2}(-[a-z]+)+-\d{1}:[0-9]{12}:group/)?[a-zA-Z0-9_\.-]{1,128}`

Required: No

GroupName (p. 12)

Deprecated - don't use this parameter. Use `Group` instead.

Type: String

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

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

Required: No

Response Syntax

```
HTTP/1.1 200
```

```
Content-type: application/json

{
   "Group": {
      "Description": "string",
      "GroupArn": "string",
      "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.

Group (p. 12)

A full description of the deleted resource group.

Type: Group (p. 80) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 111).

BadRequestException

The request includes one or more parameters that violate validation rules.

HTTP Status Code: 400

ForbiddenException

The caller isn't authorized to make the request. Check permissions.

HTTP Status Code: 403

InternalServerErrorException

An internal error occurred while processing the request. Try again later.

HTTP Status Code: 500

MethodNotAllowedException

The request uses an HTTP method that isn't allowed for the specified resource.

HTTP Status Code: 405

NotFoundException

One or more of the specified resources don't exist.

HTTP Status Code: 404

TooManyRequestsException

You've exceeded throttling limits by making too many requests in a period of time.

HTTP Status Code: 429

Examples

Example

The following example deletes the resource group with the name `MyGroup` in the `us-west-2` Region of the calling account.

Sample Request

```
POST /delete-group HTTP/1.1
Host: resource-groups.us-west-2.amazonaws.com
Accept-Encoding: identity
User-Agent: aws-cli/2.2.40 Python/3.8.8 Windows/10 exe/AMD64 prompt/off command/resource-
groups.delete-group
X-Amz-Date: 20220113T205913Z
X-Amz-Security-Token: <SECURITY-TOKEN>
Authorization: AWS4-HMAC-SHA256 Credential=<ACCESS-KEY>/20220113/us-west-2/resource-groups/
aws4_request,SignedHeaders=host;x-amz-date;x-amz-security-token,Signature=<SIGV4-SIGNATURE>
Content-Length: 20

{
    "Group": "MyGroup"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: Thu, 13 Jan 2022 20:59:14 GMT
Content-Type: application/json
Content-Length: 127
x-amzn-RequestId: <VARIES>
x-amz-apigw-id: <VARIES>
X-Amzn-Trace-Id: Root=<VARIES>
Connection: keep-alive

{
    "Group":{
        "GroupArn":"arn:aws:resource-groups:us-west-2:123456789012:group/MyGroup",
        "Name":"MyGroup",
        "OwnerId":"123456789012"
    }
}
```

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

GetGroup

Returns information about a specified resource group.

Minimum permissions

To run this command, you must have the following permissions:

- `resource-groups:GetGroup`

Request Syntax

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

{
   "Group": "string",
   "GroupName": "string"
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

Group (p. 16)

The name or the ARN of the resource group to retrieve.

Type: String

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

Pattern: `(arn:aws(-[a-z]+)*:resource-groups:[a-z]{2}(-[a-z]+)+-\d{1}:[0-9]{12}:group/)?[a-zA-Z0-9_\.-]{1,128}`

Required: No

GroupName (p. 16)

Deprecated - don't use this parameter. Use `Group` instead.

Type: String

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

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

Required: No

Response Syntax

```
HTTP/1.1 200
```

```
Content-type: application/json

{
   "Group": {
      "Description": "string",
      "GroupArn": "string",
      "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.

Group (p. 16)

A structure that contains the metadata details for the specified resource group. Use GetGroupQuery (p. 26) and GetGroupConfiguration (p. 20) to get those additional details of the resource group.

Type: Group (p. 80) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 111).

BadRequestException

The request includes one or more parameters that violate validation rules.

HTTP Status Code: 400

ForbiddenException

The caller isn't authorized to make the request. Check permissions.

HTTP Status Code: 403

InternalServerErrorException

An internal error occurred while processing the request. Try again later.

HTTP Status Code: 500

MethodNotAllowedException

The request uses an HTTP method that isn't allowed for the specified resource.

HTTP Status Code: 405

NotFoundException

One or more of the specified resources don't exist.

HTTP Status Code: 404

TooManyRequestsException

You've exceeded throttling limits by making too many requests in a period of time.

HTTP Status Code: 429

Examples

Example

The following example displays the metadata about a resource group in the `us-west-2` Region for the calling AWS account. You can then use GetGroupQuery (p. 26) and GetGroupConfiguration (p. 20) to get the details of the resource group.

Sample Request

```
POST /get-group HTTP/1.1
Host: resource-groups.us-west-2.amazonaws.com
Accept-Encoding: identity
User-Agent: aws-cli/2.2.40 Python/3.8.8 Windows/10 exe/AMD64 prompt/off command/resource-
groups.get-group
X-Amz-Date: 20220113T211658Z
X-Amz-Security-Token: <SECURITY-TOKEN>
Authorization: AWS4-HMAC-SHA256 Credential=<ACCESS-KEY>/20220113/us-west-2/resource-groups/
aws4_request,SignedHeaders=host;x-amz-date;x-amz-security-token,Signature=<SIGV4-SIGNATURE>
Content-Length: 21

{"Group": "CRPGroup"}
```

Sample Response

```
HTTP/1.1 200 OK
Date: Thu, 13 Jan 2022 21:16:58 GMT
Content-Type: application/json
Content-Length: 193
x-amzn-RequestId: <VARIES>
x-amz-apigw-id: <VARIES>
X-Amzn-Trace-Id: Root=<VARIES>
Connection: keep-alive

{
    "Group":{
        "GroupArn":"arn:aws:resource-groups:us-west-2:123456789012:group/CRPGroup",
        "Name":"CRPGroup",
        "Description":"Resource group for capacity reservations.",
        "OwnerId":"123456789012"
    }
}
```

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

GetGroupConfiguration

Retrieves the service configuration associated with the specified resource group. For details about the service configuration syntax, see Service configurations for Resource Groups.

Minimum permissions

To run this command, you must have the following permissions:

- `resource-groups:GetGroupConfiguration`

Request Syntax

```
POST /get-group-configuration HTTP/1.1
Content-type: application/json

{
   "Group": "string"
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

Group (p. 20)

The name or the ARN of the resource group for which you want to retrive the service configuration.

Type: String

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

Pattern: `(arn:aws(-[a-z]+)*:resource-groups:[a-z]{2}(-[a-z]+)+-\d{1}:[0-9]{12}:group/)?[a-zA-Z0-9_\.-]{1,128}`

Required: No

Response Syntax

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

{
   "GroupConfiguration": {
      "Configuration": [
         {
            "Parameters": [
               {
                  "Name": "string",
                  "Values": [ "string" ]
               }
```

```
            ],
            "Type": "string"
         }
      ],
      "FailureReason": "string",
      "ProposedConfiguration": [
         {
            "Parameters": [
               {
                  "Name": "string",
                  "Values": [ "string" ]
               }
            ],
            "Type": "string"
         }
      ],
      "Status": "string"
   }
}
```

Response Elements

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

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

GroupConfiguration (p. 20)

A structure that describes the service configuration attached with the specified group. For details about the service configuration syntax, see Service configurations for Resource Groups.

Type: GroupConfiguration (p. 82) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 111).

BadRequestException

The request includes one or more parameters that violate validation rules.

HTTP Status Code: 400

ForbiddenException

The caller isn't authorized to make the request. Check permissions.

HTTP Status Code: 403

InternalServerErrorException

An internal error occurred while processing the request. Try again later.

HTTP Status Code: 500

MethodNotAllowedException

The request uses an HTTP method that isn't allowed for the specified resource.

HTTP Status Code: 405

NotFoundException

One or more of the specified resources don't exist.

HTTP Status Code: 404

TooManyRequestsException

You've exceeded throttling limits by making too many requests in a period of time.

HTTP Status Code: 429

Examples

Example

The following example shows the typical output of a resource group that contains capacity reservations.

Sample Request

```
POST /get-group-configuration HTTP/1.1
Host: resource-groups.us-west-2.amazonaws.com
Accept-Encoding: identity
User-Agent: aws-cli/2.2.40 Python/3.8.8 Windows/10 exe/AMD64 prompt/off command/resource-
groups.get-group-configuration
X-Amz-Date: 20220113T213346Z
X-Amz-Security-Token: <SECURITY-TOKEN>
Authorization: AWS4-HMAC-SHA256 Credential=<ACCESS-KEY>/20220113/us-west-2/resource-groups/
aws4_request,SignedHeaders=host;x-amz-date;x-amz-security-token,Signature=<SIGV4-SIGNATURE>
Content-Length: 21

{
    "Group": "CRPGroup"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: Thu, 13 Jan 2022 21:33:46 GMT
Content-Type: application/json
Content-Length: 461
x-amzn-RequestId: <VARIES>
x-amz-apigw-id: <VARIES>
X-Amzn-Trace-Id: Root=<VARIES>
Connection: keep-alive

{
    "GroupIdentifier":{
        "GroupName":"CRPGroup",
        "GroupArn":"arn:aws:resource-groups:us-west-2:123456789012:group/CRPGroup"
    },
    "GroupConfiguration":{
        "GroupParameters":[
            {
                "Name":"allowed-resource-types",
                "Values":[
                    "AWS::EC2::CapacityReservation"
                ]
            }
        ],
        "Configuration":[
            {
                "Type":"AWS::EC2::CapacityReservationPool"
            },
```

```
            {
                "Type":"AWS::ResourceGroups::Generic",
                "Parameters":[
                    {
                        "Name":"allowed-resource-types",
                        "Values":[
                            "AWS::EC2::CapacityReservation"
                        ]
                    }
                ]
            }
        ],
        "Status":"UPDATE_COMPLETE"
    }
}
```

Example

The following example shows the typical output of a resource group that contains a configuration for Amazon EC2 dedicated hosts.

Sample Request

```
POST /get-group-configuration HTTP/1.1
Host: resource-groups.us-west-2.amazonaws.com
Accept-Encoding: identity
User-Agent: aws-cli/2.2.40 Python/3.8.8 Windows/10 exe/AMD64 prompt/off command/resource-
groups.get-group-configuration
X-Amz-Date: 20220113T213500Z
X-Amz-Security-Token: <SECURITY-TOKEN>
Authorization: AWS4-HMAC-SHA256 Credential=<ACCESS-KEY>/20220113/us-west-2/resource-groups/
aws4_request,SignedHeaders=host;x-amz-date;x-amz-security-token,Signature=<SIGV4-SIGNATURE>
Content-Length: 32

{
    "Group": "HostManagementGroup"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: Thu, 13 Jan 2022 21:35:00 GMT
Content-Type: application/json
Content-Length: 871
x-amzn-RequestId: <VARIES>
x-amz-apigw-id: <VARIES>
X-Amzn-Trace-Id: Root=<VARIES>
Connection: keep-alive

{
    "GroupIdentifier":{
        "GroupName":"HostManagementGroup",
        "GroupArn":"arn:aws:resource-groups:us-west-2:123456789012:group/
HostManagementGroup"
    },
    "GroupConfiguration":{
        "GroupParameters":[
            {
                "Name":"allowed-resource-types",
                "Values":[
                    "AWS::EC2::Host"
                ]
```

```
            },
            {
                "Name":"deletion-protection",
                "Values":[
                    "UNLESS_EMPTY"
                ]
            }
        ],
        "ResourceTypeParameters":[
            {
                "ResourceType":"AWS::EC2::Host",
                "Name":"exclusive-membership",
                "Values":[
                    "ACROSS_SAME_SERVICE_LINK_TYPE"
                ]
            }
        ],
        "ServiceLinkParameters":[
            {
                "Name":"any-host-based-license-configuration",
                "Values":[
                    "true"
                ]
            }
        ],
        "Configuration":[
            {
                "Type":"AWS::EC2::HostManagement",
                "Parameters":[
                    {
                        "Name":"any-host-based-license-configuration",
                        "Values":[
                            "true"
                        ]
                    }
                ]
            },
            {
                "Type":"AWS::ResourceGroups::Generic",
                "Parameters":[
                    {
                        "Name":"allowed-resource-types",
                        "Values":[
                            "AWS::EC2::Host"
                        ]
                    },
                    {
                        "Name":"deletion-protection",
                        "Values":[
                            "UNLESS_EMPTY"
                        ]
                    }
                ]
            }
        ],
        "Status":"UPDATE_COMPLETE"
    }
}
```


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

GetGroupQuery

Retrieves the resource query associated with the specified resource group. For more information about resource queries, see Create a tag-based group in Resource Groups.

Minimum permissions

To run this command, you must have the following permissions:

- `resource-groups:GetGroupQuery`

Request Syntax

```
POST /get-group-query HTTP/1.1
Content-type: application/json

{
   "Group": "string",
   "GroupName": "string"
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

Group (p. 26)

The name or the ARN of the resource group to query.

Type: String

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

Pattern: `(arn:aws(-[a-z]+)*:resource-groups:[a-z]{2}(-[a-z]+)+-\d{1}:[0-9]{12}:group/)?[a-zA-Z0-9_\.-]{1,128}`

Required: No

GroupName (p. 26)

This parameter has been deprecated.

Don't use this parameter. Use `Group` instead.

Type: String

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

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

Required: No

Response Syntax

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

{
   "GroupQuery": {
      "GroupName": "string",
      "ResourceQuery": {
         "Query": "string",
         "Type": "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.

GroupQuery (p. 27)

The resource query associated with the specified group. For more information about resource queries, see Create a tag-based group in Resource Groups.

Type: GroupQuery (p. 88) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 111).

BadRequestException

The request includes one or more parameters that violate validation rules.

HTTP Status Code: 400

ForbiddenException

The caller isn't authorized to make the request. Check permissions.

HTTP Status Code: 403

InternalServerErrorException

An internal error occurred while processing the request. Try again later.

HTTP Status Code: 500

MethodNotAllowedException

The request uses an HTTP method that isn't allowed for the specified resource.

HTTP Status Code: 405

NotFoundException

One or more of the specified resources don't exist.

HTTP Status Code: 404

TooManyRequestsException

You've exceeded throttling limits by making too many requests in a period of time.

HTTP Status Code: 429

Examples

Example

This example illustrates one usage of GetGroupQuery.

Sample Request

```
POST /get-group-query HTTP/1.1
Host: resource-groups.us-west-2.amazonaws.com
Accept-Encoding: identity
User-Agent: aws-cli/2.2.40 Python/3.8.8 Windows/10 exe/AMD64 prompt/off command/resource-
groups.get-group-query
X-Amz-Date: 20220114T180718Z
X-Amz-Security-Token: <SECURITY-TOKEN>
Authorization: AWS4-HMAC-SHA256 Credential=<ACCESS-KEY>/20220113/us-west-2/resource-groups/
aws4_request,SignedHeaders=host;x-amz-date;x-amz-security-token,Signature=<SIGV4-SIGNATURE>
Content-Length: 19

{
    "Group": "MyTagQueryGroup"
    }
```

Sample Response

```
HTTP/1.1 200 OK
Date: Fri, 14 Jan 2022 18:07:18 GMT
Content-Type: application/json
Content-Length: 206
x-amzn-RequestId: <VARIES>
x-amz-apigw-id: <VARIES>
X-Amzn-Trace-Id: Root=<VARIES>
Connection: keep-alive

{
    "GroupQuery":{
        "GroupName":"MyTagQueryGroup",
        "ResourceQuery":{
            "Type":"TAG_FILTERS_1_0",
            "Query":"{\"ResourceTypeFilters\":[\"AWS::EC2::Instance\"],\"TagFilters\":
[{\"Key\":\"Name\", \"Values\":[\"WebServers\"]}]}"
        }
    }
}
```

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

GetTags

Returns a list of tags that are associated with a resource group, specified by an ARN.

Minimum permissions

To run this command, you must have the following permissions:

- `resource-groups:GetTags`

Request Syntax

```
GET /resources/Arn/tags HTTP/1.1
```

URI Request Parameters

The request uses the following URI parameters.

Arn (p. 30)

The ARN of the resource group whose tags you want to retrieve.

Length Constraints: Minimum length of 12. Maximum length of 1600.

Pattern: `arn:aws(-[a-z]+)*:resource-groups:[a-z]{2}(-[a-z]+)+-\d{1}:[0-9]{12}:group/[a-zA-Z0-9_\.-]{1,128}`

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

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

{
   "Arn": "string",
   "Tags": {
      "string" : "string"
   }
}
```

Response Elements

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

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

Arn (p. 30)

The ARN of the tagged resource group.

Type: String

Length Constraints: Minimum length of 12. Maximum length of 1600.

Pattern: `arn:aws(-[a-z]+)*:resource-groups:[a-z]{2}(-[a-z]+)+-\d{1}:[0-9]{12}:group/[a-zA-Z0-9_\.-]{1,128}`

Tags (p. 30)

The tags associated with the specified resource group.

Type: String to string map

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

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

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

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 111).

BadRequestException

The request includes one or more parameters that violate validation rules.

HTTP Status Code: 400

ForbiddenException

The caller isn't authorized to make the request. Check permissions.

HTTP Status Code: 403

InternalServerErrorException

An internal error occurred while processing the request. Try again later.

HTTP Status Code: 500

MethodNotAllowedException

The request uses an HTTP method that isn't allowed for the specified resource.

HTTP Status Code: 405

NotFoundException

One or more of the specified resources don't exist.

HTTP Status Code: 404

TooManyRequestsException

You've exceeded throttling limits by making too many requests in a period of time.

HTTP Status Code: 429

Examples

Example

The following example retrieves the tags attached to the resource group with this ARN:

`arn:aws:resource-groups:us-east-1:123456789012:group/MyGroup/tags`

by querying the service's endpoint in `us-east-1`. Note that the ARN must be URL encoded.

Sample Request

```
GET /resources/arn%3Aaws%3Aresource-groups%3Aus-east-1%3A123456789012%3Agroup%2FMyGroup/
tags HTTP/1.1
Host: resource-groups.us-east-1.amazonaws.com
Accept-Encoding: identity
User-Agent: aws-cli/2.2.40 Python/3.8.8 Windows/10 exe/AMD64 prompt/off command/resource-
groups.get-tags
X-Amz-Date: 20211201T173340Z
X-Amz-Security-Token: <SECURITY-TOKEN>
Authorization: AWS4-HMAC-SHA256 Credential=<ACCESS-KEY>/20220113/us-west-2/resource-groups/
aws4_request,SignedHeaders=host;x-amz-date;x-amz-security-token,Signature=<SIGV4-SIGNATURE>
```

Sample Response

```
HTTP/1.1 200 OK
Date: Fri, 14 Jan 2022 18:40:33 GMT
Content-Type: application/json
Content-Length: 97
x-amzn-RequestId: <VARIES>
x-amz-apigw-id: <VARIES>
X-Amzn-Trace-Id: Root=<VARIES>
Connection: keep-alive

{
    "Arn": "arn:aws:resource-groups:us-east-1:123456789012:group/MyGroup",
    "Tags":{
        "Stage":"Production"
    }
}
```

See Also

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

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

GroupResources

Adds the specified resources to the specified group.

Important
You can use this operation with only resource groups that are configured with the following types:

- `AWS::EC2::HostManagement`
- `AWS::EC2::CapacityReservationPool`

Other resource group type and resource types aren't currently supported by this operation.

Minimum permissions

To run this command, you must have the following permissions:

- `resource-groups:GroupResources`

Request Syntax

```
POST /group-resources HTTP/1.1
Content-type: application/json

{
   "Group": "string",
   "ResourceArns": [ "string" ]
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

Group (p. 33)

The name or the ARN of the resource group to add resources to.

Type: String

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

Pattern: `(arn:aws(-[a-z]+)*:resource-groups:[a-z]{2}(-[a-z]+)+-\d{1}:[0-9]{12}:group/)?[a-zA-Z0-9_\.-]{1,128}`

Required: Yes

ResourceArns (p. 33)

The list of ARNs of the resources to be added to the group.

Type: Array of strings

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

Pattern: `arn:aws(-[a-z]+)*:[a-z0-9\-]*:([a-z]{2}(-[a-z]+)+-\d{1})?:([0-9]{12})?:.+`

Required: Yes

Response Syntax

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

{
   "Failed": [
      {
         "ErrorCode": "string",
         "ErrorMessage": "string",
         "ResourceArn": "string"
      }
   ],
   "Pending": [
      {
         "ResourceArn": "string"
      }
   ],
   "Succeeded": [ "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.

Failed (p. 34)

A list of ARNs of any resources that this operation failed to add to the group.

Type: Array of FailedResource (p. 79) objects

Pending (p. 34)

A list of ARNs of any resources that this operation is still in the process adding to the group. These pending additions continue asynchronously. You can check the status of pending additions by using the `ListGroupResources (p. 37)` operation, and checking the `Resources` array in the response and the `Status` field of each object in that array.

Type: Array of PendingResource (p. 90) objects

Succeeded (p. 34)

A list of ARNs of the resources that this operation successfully added to the group.

Type: Array of strings

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

Pattern: `arn:aws(-[a-z]+)*:[a-z0-9\-]*:([a-z]{2}(-[a-z]+)+-\d{1})?:([0-9]{12})?:.+`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 111).

BadRequestException

The request includes one or more parameters that violate validation rules.

HTTP Status Code: 400

ForbiddenException

The caller isn't authorized to make the request. Check permissions.

HTTP Status Code: 403

InternalServerErrorException

An internal error occurred while processing the request. Try again later.

HTTP Status Code: 500

MethodNotAllowedException

The request uses an HTTP method that isn't allowed for the specified resource.

HTTP Status Code: 405

NotFoundException

One or more of the specified resources don't exist.

HTTP Status Code: 404

TooManyRequestsException

You've exceeded throttling limits by making too many requests in a period of time.

HTTP Status Code: 429

Examples

Example

The following example adds two Amazon EC2 capacity reservations to the specified resource group. The group is configured to allow only capacity reservations as group members.

Sample Request

```
POST /group-resources HTTP/1.1
Host: resource-groups.us-west-2.amazonaws.com
Accept-Encoding: identity
User-Agent: aws-cli/2.2.40 Python/3.8.8 Windows/10 exe/AMD64 prompt/off command/resource-
groups.group-resources
X-Amz-Date: 20220114T203701Z
X-Amz-Security-Token: <SECURITY-TOKEN>
Authorization: AWS4-HMAC-SHA256 Credential=<ACCESS-KEY>/20220113/us-west-2/resource-groups/
aws4_request,SignedHeaders=host;x-amz-date;x-amz-security-token,Signature=<SIGV4-SIGNATURE>
Content-Length: 199

{
    "Group": "CRPGroup",
```

```
    "ResourceArns": [
        "arn:aws:ec2:us-west-2:123456789012:capacity-reservation/cr-0070b00d13EXAMPLE",
        "arn:aws:ec2:us-west-2:123456789012:capacity-reservation/cr-061abec820EXAMPLE"
    ]
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: Fri, 14 Jan 2022 20:37:01 GMT
Content-Type: application/json
Content-Length: 119
x-amzn-RequestId: <VARIES>
x-amz-apigw-id: <VARIES>
X-Amzn-Trace-Id: Root=<VARIES>
Connection: keep-alive


{
    "Succeeded":[
        "arn:aws:ec2:us-west-2:123456789012:capacity-reservation/cr-0070b00d13EXAMPLE",
        "arn:aws:ec2:us-west-2:123456789012:capacity-reservation/cr-061abec820EXAMPLE"
    ],
    "Failed":[],
    "Pending":[]
}
```

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

ListGroupResources

Returns a list of ARNs of the resources that are members of a specified resource group.

Minimum permissions

To run this command, you must have the following permissions:

- `resource-groups:ListGroupResources`
- `cloudformation:DescribeStacks`
- `cloudformation:ListStackResources`
- `tag:GetResources`

Request Syntax

```
POST /list-group-resources HTTP/1.1
Content-type: application/json

{
   "Filters": [
      {
         "Name": "string",
         "Values": [ "string" ]
      }
   ],
   "Group": "string",
   "GroupName": "string",
   "MaxResults": number,
   "NextToken": "string"
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

Filters (p. 37)

Filters, formatted as ResourceFilter (p. 92) objects, that you want to apply to a `ListGroupResources` operation. Filters the results to include only those of the specified resource types.

- `resource-type` - Filter resources by their type. Specify up to five resource types in the format `AWS::ServiceCode::ResourceType`. For example, `AWS::EC2::Instance`, or `AWS::S3::Bucket`.

When you specify a `resource-type` filter for `ListGroupResources`, AWS Resource Groups validates your filter resource types against the types that are defined in the query associated with the group. For example, if a group contains only S3 buckets because its query specifies only that resource type, but your `resource-type` filter includes EC2 instances, AWS Resource Groups does not filter for EC2 instances. In this case, a `ListGroupResources` request returns a `BadRequestException` error with a message similar to the following:

`The resource types specified as filters in the request are not valid.`

The error includes a list of resource types that failed the validation because they are not part of the query associated with the group. This validation doesn't occur when the group query specifies `AWS::AllSupported`, because a group based on such a query can contain any of the allowed resource types for the query type (tag-based or Amazon CloudFront stack-based queries).

Type: Array of ResourceFilter (p. 92) objects

Required: No

Group (p. 37)

The name or the ARN of the resource group

Type: String

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

Pattern: `(arn:aws(-[a-z]+)*:resource-groups:[a-z]{2}(-[a-z]+)+-\d{1}:[0-9]{12}:group/)?[a-zA-Z0-9_\.-]{1,128}`

Required: No

GroupName (p. 37)

Important
Deprecated - don't use this parameter. Use the `Group` request field instead.

Type: String

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

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

Required: No

MaxResults (p. 37)

The total number of results that you want included on each page of the response. If you do not include this parameter, it defaults to a value that is specific to the operation. If additional items exist beyond the maximum you specify, the `NextToken` response element is present and has a value (is not null). Include that value as the `NextToken` request parameter in the next call to the operation to get the next part of the results. Note that the service might return fewer results than the maximum even when there are more results available. You should check `NextToken` after every operation to ensure that you receive all of the results.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 50.

Required: No

NextToken (p. 37)

The parameter for receiving additional results if you receive a `NextToken` response in a previous request. A `NextToken` response indicates that more output is available. Set this parameter to the value provided by a previous call's `NextToken` response to indicate where the output should continue from.

Type: String

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

Pattern: `^[a-zA-Z0-9+/]*={0,2}$`

Required: No

Response Syntax

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

{
   "NextToken": "string",
   "QueryErrors": [
      {
         "ErrorCode": "string",
         "Message": "string"
      }
   ],
   "ResourceIdentifiers": [
      {
         "ResourceArn": "string",
         "ResourceType": "string"
      }
   ],
   "Resources": [
      {
         "Identifier": {
            "ResourceArn": "string",
            "ResourceType": "string"
         },
         "Status": {
            "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.

NextToken (p. 39)

If present, indicates that more output is available than is included in the current response. Use this value in the `NextToken` request parameter in a subsequent call to the operation to get the next part of the output. You should repeat this until the `NextToken` response element comes back as `null`.

Type: String

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

Pattern: `^[a-zA-Z0-9+/]*={0,2}$`

QueryErrors (p. 39)

A list of `QueryError` objects. Each error is an object that contains `ErrorCode` and `Message` structures. Possible values for `ErrorCode` are `CLOUDFORMATION_STACK_INACTIVE` and `CLOUDFORMATION_STACK_NOT_EXISTING`.

Type: Array of QueryError (p. 91) objects

ResourceIdentifiers (p. 39)

Important
Deprecated - don't use this parameter. Use the `Resources` response field instead.

Type: Array of ResourceIdentifier (p. 93) objects

Resources (p. 39)

An array of resources from which you can determine each resource's identity, type, and group membership status.

Type: Array of ListGroupResourcesItem (p. 89) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 111).

BadRequestException

The request includes one or more parameters that violate validation rules.

HTTP Status Code: 400

ForbiddenException

The caller isn't authorized to make the request. Check permissions.

HTTP Status Code: 403

InternalServerErrorException

An internal error occurred while processing the request. Try again later.

HTTP Status Code: 500

MethodNotAllowedException

The request uses an HTTP method that isn't allowed for the specified resource.

HTTP Status Code: 405

NotFoundException

One or more of the specified resources don't exist.

HTTP Status Code: 404

TooManyRequestsException

You've exceeded throttling limits by making too many requests in a period of time.

HTTP Status Code: 429

UnauthorizedException

The request was rejected because it doesn't have valid credentials for the target resource.

HTTP Status Code: 401

Examples

Example

This example illustrates one usage of ListGroupResources.

Sample Request

```
POST /list-group-resources HTTP/1.1
Host: resource-groups.us-west-2.amazonaws.com
Accept-Encoding: identity
User-Agent: aws-cli/2.2.40 Python/3.8.8 Windows/10 exe/AMD64 prompt/off command/resource-
groups.list-group-resources
X-Amz-Date: 20220114T205755Z
X-Amz-Security-Token: <SECURITY-TOKEN>
Authorization: AWS4-HMAC-SHA256 Credential=<ACCESS-KEY>/20220113/us-west-2/resource-groups/
aws4_request,SignedHeaders=host;x-amz-date;x-amz-security-token,Signature=<SIGV4-SIGNATURE>
Content-Length: 21

{
    "Group": "CRPGroup"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: Fri, 14 Jan 2022 20:57:55 GMT
Content-Type: application/json
Content-Length: 654
x-amzn-RequestId: <VARIES>
x-amz-apigw-id: <VARIES>
X-Amzn-Trace-Id: Root=<VARIES>
Connection: keep-alive

{
    "Resources":[
        {
            "Identifier":{
                "ResourceArn":"arn:aws:ec2:us-west-2:123456789012:capacity-reservation/
cr-0070b00d13EXAMPLE",
                "ResourceType":"AWS::EC2::CapacityReservation"
            }
        },
        {
            "Identifier":{
                "ResourceArn":"arn:aws:ec2:us-west-2:123456789012:capacity-reservation/
cr-061abec820EXAMPLE",
                "ResourceType":"AWS::EC2::CapacityReservation"
            }
        }
    ],
    "QueryErrors":[]
}
```

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

ListGroups

Returns a list of existing Resource Groups in your account.

Minimum permissions

To run this command, you must have the following permissions:

- `resource-groups:ListGroups`

Request Syntax

```
POST /groups-list?maxResults=MaxResults&nextToken=NextToken HTTP/1.1
Content-type: application/json

{
   "Filters": [
      {
         "Name": "string",
         "Values": [ "string" ]
      }
   ]
}
```

URI Request Parameters

The request uses the following URI parameters.

MaxResults (p. 43)

The total number of results that you want included on each page of the response. If you do not include this parameter, it defaults to a value that is specific to the operation. If additional items exist beyond the maximum you specify, the `NextToken` response element is present and has a value (is not null). Include that value as the `NextToken` request parameter in the next call to the operation to get the next part of the results. Note that the service might return fewer results than the maximum even when there are more results available. You should check `NextToken` after every operation to ensure that you receive all of the results.

Valid Range: Minimum value of 1. Maximum value of 50.

NextToken (p. 43)

The parameter for receiving additional results if you receive a `NextToken` response in a previous request. A `NextToken` response indicates that more output is available. Set this parameter to the value provided by a previous call's `NextToken` response to indicate where the output should continue from.

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

Pattern: `^[a-zA-Z0-9+/]*={0,2}$`

Request Body

The request accepts the following data in JSON format.

Filters (p. 43)

Filters, formatted as GroupFilter (p. 86) objects, that you want to apply to a `ListGroups` operation.

- `resource-type` - Filter the results to include only those of the specified resource types. Specify up to five resource types in the format `AWS::`*`ServiceCode`*`::`*`ResourceType`* . For example, `AWS::EC2::Instance`, or `AWS::S3::Bucket`.
- `configuration-type` - Filter the results to include only those groups that have the specified configuration types attached. The current supported values are:
  - `AWS::EC2::CapacityReservationPool`
  - `AWS::EC2::HostManagement`

Type: Array of GroupFilter (p. 86) objects

Required: No

Response Syntax

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

{
   "GroupIdentifiers": [
      {
         "GroupArn": "string",
         "GroupName": "string"
      }
   ],
   "Groups": [
      {
         "Description": "string",
         "GroupArn": "string",
         "Name": "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.

GroupIdentifiers (p. 44)

A list of GroupIdentifier (p. 87) objects. Each identifier is an object that contains both the `Name` and the `GroupArn`.

Type: Array of GroupIdentifier (p. 87) objects

Groups (p. 44)

Important
Deprecated - don't use this field. Use the `GroupIdentifiers` response field instead.

Type: Array of Group (p. 80) objects

NextToken (p. 44)

If present, indicates that more output is available than is included in the current response. Use this value in the `NextToken` request parameter in a subsequent call to the operation to get the next part of the output. You should repeat this until the `NextToken` response element comes back as `null`.

Type: String

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

Pattern: `^[a-zA-Z0-9+/]*={0,2}$`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 111).

BadRequestException

The request includes one or more parameters that violate validation rules.

HTTP Status Code: 400

ForbiddenException

The caller isn't authorized to make the request. Check permissions.

HTTP Status Code: 403

InternalServerErrorException

An internal error occurred while processing the request. Try again later.

HTTP Status Code: 500

MethodNotAllowedException

The request uses an HTTP method that isn't allowed for the specified resource.

HTTP Status Code: 405

TooManyRequestsException

You've exceeded throttling limits by making too many requests in a period of time.

HTTP Status Code: 429

Examples

Example

The following example lists all of the resource groups in the specified AWS Region and account.

Sample Request

```
POST /groups-list HTTP/1.1
Host: resource-groups.us-west-2.amazonaws.com
Accept-Encoding: identity
User-Agent: aws-cli/2.2.40 Python/3.8.8 Windows/10 exe/AMD64 prompt/off command/resource-
groups.list-groups
X-Amz-Date: 20220119T231526Z
```

```
X-Amz-Security-Token: <SECURITY-TOKEN>
Authorization: AWS4-HMAC-SHA256 Credential=<ACCESS-KEY>/20220113/us-west-2/resource-groups/
aws4_request,SignedHeaders=host;x-amz-date;x-amz-security-token,Signature=<SIGV4-SIGNATURE>
Content-Length: 0
```

Sample Response

```
HTTP/1.1 200 OK
Date: Wed, 19 Jan 2022 23:15:26 GMT
Content-Type: application/json
Content-Length: 5655
x-amzn-RequestId: <VARIES>
x-amz-apigw-id: <VARIES>
X-Amzn-Trace-Id: Root=<VARIES>
Connection: keep-alive

{
    "GroupIdentifiers": [
        {
            "GroupName": "Group1",
            "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/Group1",
            "OwnerId": "123456789012"
        },
        {
            "GroupName": "Group2",
            "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/Group2",
            "OwnerId": "123456789012"
        },
        {
            "GroupName": "Group3",
            "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/Group3",
            "OwnerId": "123456789012"
        }
    ],
    "Groups": [  ... DEPRECATED ... DO NOT USE ... ]
}
```

Example

The following example lists all resource groups that are configured with a configure type of `AWS::EC2::HostMangement`.

Sample Request

```
POST /groups-list HTTP/1.1
Host: resource-groups.us-west-2.amazonaws.com
Accept-Encoding: identity
User-Agent: aws-cli/2.2.40 Python/3.8.8 Windows/10 exe/AMD64 prompt/off command/resource-
groups.list-groups
X-Amz-Date: 20220119T232648Z
X-Amz-Security-Token: <SECURITY-TOKEN>
Authorization: AWS4-HMAC-SHA256 Credential=<ACCESS-KEY>/20220113/us-west-2/resource-groups/
aws4_request,SignedHeaders=host;x-amz-date;x-amz-security-token,Signature=<SIGV4-SIGNATURE>
Content-Length: 85

{
    "Filters": [
        {
            "Name": "configuration-type",
            "Values": [
                "AWS::EC2::HostManagement"
            ]
```

```
        }
    ]
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: Wed, 19 Jan 2022 23:26:48 GMT
Content-Type: application/json
Content-Length: 621
x-amzn-RequestId: <VARIES>
x-amz-apigw-id: <VARIES>
X-Amzn-Trace-Id: Root=<VARIES>
Connection: keep-alive

{
    "GroupIdentifiers": [
        {
            "GroupName": "HostManagementGroup1",
            "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/
HostManagementGroup1",
            "OwnerId": "123456789012"
        },
        {
            "GroupName": "HostManagementGroup2",
            "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/
HostManagementGroup2",
            "OwnerId": "123456789012"
        }
    ],
    "Groups": [ ... DEPRECATED ... DO NOT USE ... ]
}
```

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

PutGroupConfiguration

Attaches a service configuration to the specified group. This occurs asynchronously, and can take time to complete. You can use GetGroupConfiguration (p. 20) to check the status of the update.

Minimum permissions

To run this command, you must have the following permissions:

- `resource-groups:PutGroupConfiguration`

Request Syntax

```
POST /put-group-configuration HTTP/1.1
Content-type: application/json

{
   "Configuration": [
      {
         "Parameters": [
            {
               "Name": "string",
               "Values": [ "string" ]
            }
         ],
         "Type": "string"
      }
   ],
   "Group": "string"
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

Configuration (p. 48)

The new configuration to associate with the specified group. A configuration associates the resource group with an AWS service and specifies how the service can interact with the resources in the group. A configuration is an array of GroupConfigurationItem (p. 84) elements.

For information about the syntax of a service configuration, see Service configurations for Resource Groups.

Note
A resource group can contain either a `Configuration` or a `ResourceQuery`, but not both.

Type: Array of GroupConfigurationItem (p. 84) objects

Array Members: Maximum number of 2 items.

Required: No

Group (p. 48)

The name or ARN of the resource group with the configuration that you want to update.

Type: String

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

Pattern: `(arn:aws(-[a-z]+)*:resource-groups:[a-z]{2}(-[a-z]+)+-\d{1}:[0-9]{12}:group/)?[a-zA-Z0-9_\.-]{1,128}`

Required: No

Response Syntax

```
HTTP/1.1 202
```

Response Elements

If the action is successful, the service sends back an HTTP 202 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 111).

BadRequestException

The request includes one or more parameters that violate validation rules.

HTTP Status Code: 400

ForbiddenException

The caller isn't authorized to make the request. Check permissions.

HTTP Status Code: 403

InternalServerErrorException

An internal error occurred while processing the request. Try again later.

HTTP Status Code: 500

MethodNotAllowedException

The request uses an HTTP method that isn't allowed for the specified resource.

HTTP Status Code: 405

NotFoundException

One or more of the specified resources don't exist.

HTTP Status Code: 404

TooManyRequestsException

You've exceeded throttling limits by making too many requests in a period of time.

HTTP Status Code: 429

Examples

Example 1: Configure a resource group to contain Amazon EC2 capacity reservations

The following example attaches a configuration that limits the group to containing Amazon EC2 capacity reservations.

Sample Request

```
POST /put-group-configuration HTTP/1.1
Host: resource-groups.us-west-2.amazonaws.com
Accept-Encoding: identity
User-Agent: aws-cli/2.2.40 Python/3.8.8 Windows/10 exe/AMD64 prompt/off command/resource-
groups.put-group-configuration
X-Amz-Date: 20220120T204033Z
X-Amz-Security-Token: <SECURITY-TOKEN>
Authorization: AWS4-HMAC-SHA256 Credential=<ACCESS-KEY>/20220113/us-west-2/resource-groups/
aws4_request,SignedHeaders=host;x-amz-date;x-amz-security-token,Signature=<SIGV4-SIGNATURE>
Content-Length: 226

{
    "Group": "CRPGroup",
    "Configuration": [
        {
            "Type": "AWS::EC2::CapacityReservationPool"
        },
        {
            "Type": "AWS::ResourceGroups::Generic",
            "Parameters": [
                {
                    "Name": "allowed-resource-types",
                    "Values": [
                        "AWS::EC2::CapacityReservation"
                    ]
                }
            ]
        }
    ]
}
```

Sample Response

```
HTTP/1.1 202 Accepted
Date: Thu, 20 Jan 2022 20:40:33 GMT
Content-Type: application/json
Content-Length: 337
x-amzn-RequestId: <VARIES>
x-amz-apigw-id: <VARIES>
X-Amzn-Trace-Id: Root=<VARIES>
Connection: keep-alive

{
    "GroupIdentifier": {
        "GroupName": "CRPGroup",
        "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/CRPGroup"
    },
    "GroupConfiguration": {
        "Configuration": [
            {
```

```
                "Type": "AWS::EC2::CapacityReservationPool"
            },
            {
                "Type": "AWS::ResourceGroups::Generic",
                "Parameters": [
                    {
                        "Name": "allowed-resource-types",
                        "Values": [
                            "AWS::EC2::CapacityReservation"
                        ]
                    }
                ]
            }
        ]
    }
}
```

Example 2: Configure a resource group to contain Amazon EC2 hosts of family type C5 with license support

The following example attaches a configuration to the resource group that limits the group to containing only Amazon EC2 hosts and provides settings that are enforced on any Amazon EC2 instances that are launched into this group.

Sample Request

```
POST /put-group-configuration HTTP/1.1
Host: resource-groups.us-west-2.amazonaws.com
Accept-Encoding: identity
User-Agent: aws-cli/2.2.40 Python/3.8.8 Windows/10 exe/AMD64 prompt/off command/resource-
groups.put-group-configuration
X-Amz-Date: 20220120T204033Z
X-Amz-Security-Token: <SECURITY-TOKEN>
Authorization: AWS4-HMAC-SHA256 Credential=<ACCESS-KEY>/20220113/us-west-2/resource-groups/
aws4_request,SignedHeaders=host;x-amz-date;x-amz-security-token,Signature=<SIGV4-SIGNATURE>
Content-Length: 314

{
    "Group": "HostManagementGroup",
    "Configuration": [
        {
          "Type": "AWS::EC2::HostManagement",
          "Parameters": [
            {
              "Name":"any-host-based-license-configuration",
              "Values":["true"]
            },
            {
              "Name":"allowed-host-families",
              "Values":["c5"]
            }
          ]
        },
        {
          "Type": "AWS::ResourceGroups::Generic",
          "Parameters": [
            {
              "Name": "allowed-resource-types",
              "Values": ["AWS::EC2::Host"]
            },
            {
              "Name": "deletion-protection",
```

```
              "Values": ["UNLESS_EMPTY"]
            }
          ]
        }
    ]
}
```

Sample Response

```
HTTP/1.1 202 Accepted
Date: Thu, 20 Jan 2022 20:40:33 GMT
Content-Type: application/json
Content-Length: 337
x-amzn-RequestId: <VARIES>
x-amz-apigw-id: <VARIES>
X-Amzn-Trace-Id: Root=<VARIES>
Connection: keep-alive

{
    "GroupIdentifier": {
        "GroupName": "HostManagementGroup",
        "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/
HostManagementGroup"
    },
    "GroupConfiguration": {
        "Configuration": [
            {
                "Type": "AWS::EC2::HostManagement",
                "Parameters": [
                    {
                        "Name":"any-host-based-license-configuration",
                        "Values":["true"]
                    },
                    {
                        "Name":"allowed-host-families",
                        "Values":["c5"]
                    }
                ]
            },
            {
              "Type": "AWS::ResourceGroups::Generic",
              "Parameters": [
                {
                  "Name": "allowed-resource-types",
                  "Values": ["AWS::EC2::Host"]
                },
                {
                  "Name": "deletion-protection",
                  "Values": ["UNLESS_EMPTY"]
                }
              ]
            }
        ]
    }
}
```

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

SearchResources

Returns a list of AWS resource identifiers that matches the specified query. The query uses the same format as a resource query in a CreateGroup (p. 3) or UpdateGroupQuery (p. 74) operation.

Minimum permissions

To run this command, you must have the following permissions:

- `resource-groups:SearchResources`
- `cloudformation:DescribeStacks`
- `cloudformation:ListStackResources`
- `tag:GetResources`

Request Syntax

```
POST /resources/search HTTP/1.1
Content-type: application/json

{
   "MaxResults": number,
   "NextToken": "string",
   "ResourceQuery": {
      "Query": "string",
      "Type": "string"
   }
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

ResourceQuery (p. 54)

The search query, using the same formats that are supported for resource group definition. For more information, see CreateGroup (p. 3).

Type: ResourceQuery (p. 94) object

Required: Yes

MaxResults (p. 54)

The total number of results that you want included on each page of the response. If you do not include this parameter, it defaults to a value that is specific to the operation. If additional items exist beyond the maximum you specify, the `NextToken` response element is present and has a value (is not null). Include that value as the `NextToken` request parameter in the next call to the operation to get the next part of the results. Note that the service might return fewer results than the maximum even when there are more results available. You should check `NextToken` after every operation to ensure that you receive all of the results.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 50.

Required: No

NextToken (p. 54)

The parameter for receiving additional results if you receive a `NextToken` response in a previous request. A `NextToken` response indicates that more output is available. Set this parameter to the value provided by a previous call's `NextToken` response to indicate where the output should continue from.

Type: String

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

Pattern: `^[a-zA-Z0-9+/]*={0,2}$`

Required: No

Response Syntax

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

{
   "NextToken": "string",
   "QueryErrors": [
      {
         "ErrorCode": "string",
         "Message": "string"
      }
   ],
   "ResourceIdentifiers": [
      {
         "ResourceArn": "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 (p. 55)

If present, indicates that more output is available than is included in the current response. Use this value in the `NextToken` request parameter in a subsequent call to the operation to get the next part of the output. You should repeat this until the `NextToken` response element comes back as `null`.

Type: String

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

Pattern: `^[a-zA-Z0-9+/]*={0,2}$`

QueryErrors (p. 55)

A list of `QueryError` objects. Each error is an object that contains `ErrorCode` and `Message` structures.

Possible values for `ErrorCode`:

- `CLOUDFORMATION_STACK_INACTIVE`
- `CLOUDFORMATION_STACK_NOT_EXISTING`

Type: Array of QueryError (p. 91) objects

ResourceIdentifiers (p. 55)

The ARNs and resource types of resources that are members of the group that you specified.

Type: Array of ResourceIdentifier (p. 93) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 111).

BadRequestException

The request includes one or more parameters that violate validation rules.

HTTP Status Code: 400

ForbiddenException

The caller isn't authorized to make the request. Check permissions.

HTTP Status Code: 403

InternalServerErrorException

An internal error occurred while processing the request. Try again later.

HTTP Status Code: 500

MethodNotAllowedException

The request uses an HTTP method that isn't allowed for the specified resource.

HTTP Status Code: 405

TooManyRequestsException

You've exceeded throttling limits by making too many requests in a period of time.

HTTP Status Code: 429

UnauthorizedException

The request was rejected because it doesn't have valid credentials for the target resource.

HTTP Status Code: 401

Examples

Example

The following example retrieves a list of resources of all types that are tagged with a key of `Stage` and a value of either `Test` or `Prod`.

Sample Request

```
POST /resources/search HTTP/1.1
Host: resource-groups.us-west-2.amazonaws.com
Accept-Encoding: identity
User-Agent: aws-cli/2.2.40 Python/3.8.8 Windows/10 exe/AMD64 prompt/off command/resource-
groups.search-resources
X-Amz-Date: 20220120T213215Z
X-Amz-Security-Token: <SECURITY-TOKEN>
Authorization: AWS4-HMAC-SHA256 Credential=<ACCESS-KEY>/20220113/us-west-2/resource-groups/
aws4_request,SignedHeaders=host;x-amz-date;x-amz-security-token,Signature=<SIGV4-SIGNATURE>
Content-Length: 176

{
    "ResourceQuery": {
        "Query": "{\"ResourceTypeFilters\":[\"AWS::AllSupported\"],\"TagFilters\":[{\"Key
\":\"Stage\",\"Values\":[\"Test\",\"Prod\"]}]}",
        "Type": "TAG_FILTERS_1_0"
    }
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: Thu, 20 Jan 2022 21:32:15 GMT
Content-Type: application/json
Content-Length: 161
x-amzn-RequestId: <VARIES>
x-amz-apigw-id: <VARIES>
X-Amzn-Trace-Id: Root=<VARIES>
Connection: keep-alive

{
    "ResourceIdentifiers": [
        {
            "ResourceArn": "arn:aws:ec2:us-west-2:123456789012:dhcp-options/dopt-1a2b3c4d",
            "ResourceType": "AWS::EC2::DHCPOptions"
        }
    ],
    "QueryErrors": []
}
```

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

Tag

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

> **Important**
> Do not store personally identifiable information (PII) or other confidential or sensitive information in tags. We use tags to provide you with billing and administration services. Tags are not intended to be used for private or sensitive data.

Minimum permissions

To run this command, you must have the following permissions:

- `resource-groups:Tag`

Request Syntax

```
PUT /resources/Arn/tags HTTP/1.1
Content-type: application/json

{
   "Tags": {
      "string" : "string"
   }
}
```

URI Request Parameters

The request uses the following URI parameters.

Arn (p. 58)

The ARN of the resource group to which to add tags.

Length Constraints: Minimum length of 12. Maximum length of 1600.

Pattern: `arn:aws(-[a-z]+)*:resource-groups:[a-z]{2}(-[a-z]+)+-\d{1}:[0-9]{12}:group/[a-zA-Z0-9_\.-]{1,128}`

Required: Yes

Request Body

The request accepts the following data in JSON format.

Tags (p. 58)

The tags to add to the specified resource group. A tag is a string-to-string map of key-value pairs.

Type: String to string map

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

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

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

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

Required: Yes

Response Syntax

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

{
   "Arn": "string",
   "Tags": {
      "string" : "string"
   }
}
```

Response Elements

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

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

Arn (p. 59)

The ARN of the tagged resource.

Type: String

Length Constraints: Minimum length of 12. Maximum length of 1600.

Pattern: `arn:aws(-[a-z]+)*:resource-groups:[a-z]{2}(-[a-z]+)+-\d{1}:[0-9]{12}:group/[a-zA-Z0-9_\.-]{1,128}`

Tags (p. 59)

The tags that have been added to the specified resource group.

Type: String to string map

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

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

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

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 111).

BadRequestException

The request includes one or more parameters that violate validation rules.

HTTP Status Code: 400

ForbiddenException

The caller isn't authorized to make the request. Check permissions.

HTTP Status Code: 403

InternalServerErrorException

An internal error occurred while processing the request. Try again later.

HTTP Status Code: 500

MethodNotAllowedException

The request uses an HTTP method that isn't allowed for the specified resource.

HTTP Status Code: 405

NotFoundException

One or more of the specified resources don't exist.

HTTP Status Code: 404

TooManyRequestsException

You've exceeded throttling limits by making too many requests in a period of time.

HTTP Status Code: 429

Examples

Example

The following example attaches two tags to the specified resource group. The first tag has a key of `Department` and a value of `Finance`, and the second has a key of `Env` and a value of `Prod`. Note that this tags only the group itself; it doesn't tag any members of the group.

The ARN in the PUT element must be URL encoded.

Sample Request

```
PUT /resources/arn%3Aaws%3Aresource-groups%3Aus-west-2%3A123456789012%3Agroup%2FMyTest/tags
 HTTP/1.1
Host: resource-groups.us-west-2.amazonaws.com
Accept-Encoding: identity
User-Agent: aws-cli/2.2.40 Python/3.8.8 Windows/10 exe/AMD64 prompt/off command/resource-
groups.tag
X-Amz-Date: 20220120T220458Z
X-Amz-Security-Token: <SECURITY-TOKEN>
Authorization: AWS4-HMAC-SHA256 Credential=<ACCESS-KEY>/20220113/us-west-2/resource-groups/
aws4_request,SignedHeaders=host;x-amz-date;x-amz-security-token,Signature=<SIGV4-SIGNATURE>
Content-Length: 50

{
    "Tags": {
        "Department": "Finance",
        "Env": "Prod"
    }
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: Thu, 20 Jan 2022 22:04:58 GMT
Content-Type: application/json
Content-Length: 114
x-amzn-RequestId: <VARIES>
x-amz-apigw-id: <VARIES>
X-Amzn-Trace-Id: Root=<VARIES>
Connection: keep-alive

{
    "Arn":"arn:aws:resource-groups:us-west-2:123456789012:group/MyTest",
    "Tags":{
        "Department":"Finance",
        "Env":"Prod"
    }
}
```

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

UngroupResources

Removes the specified resources from the specified group. This operation works only with static groups that you populated using the GroupResources (p. 33) operation. It doesn't work with any resource groups that are automatically populated by tag-based or AWS CloudFormation stack-based queries.

Minimum permissions

To run this command, you must have the following permissions:

- `resource-groups:UngroupResources`

Request Syntax

```
POST /ungroup-resources HTTP/1.1
Content-type: application/json

{
   "Group": "string",
   "ResourceArns": [ "string" ]
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

Group (p. 62)

The name or the ARN of the resource group from which to remove the resources.

Type: String

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

Pattern: `(arn:aws(-[a-z]+)*:resource-groups:[a-z]{2}(-[a-z]+)+-\d{1}:[0-9]{12}:group/)?[a-zA-Z0-9_\.-]{1,128}`

Required: Yes

ResourceArns (p. 62)

The ARNs of the resources to be removed from the group.

Type: Array of strings

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

Pattern: `arn:aws(-[a-z]+)*:[a-z0-9\-]*:([a-z]{2}(-[a-z]+)+-\d{1})?:([0-9]{12})?:.+`

Required: Yes

Response Syntax

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

{
   "Failed": [
      {
         "ErrorCode": "string",
         "ErrorMessage": "string",
         "ResourceArn": "string"
      }
   ],
   "Pending": [
      {
         "ResourceArn": "string"
      }
   ],
   "Succeeded": [ "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.

Failed (p. 63)

A list of any resources that failed to be removed from the group by this operation.

Type: Array of FailedResource (p. 79) objects

Pending (p. 63)

A list of any resources that are still in the process of being removed from the group by this operation. These pending removals continue asynchronously. You can check the status of pending removals by using the `ListGroupResources (p. 37)` operation. After the resource is successfully removed, it no longer appears in the response.

Type: Array of PendingResource (p. 90) objects

Succeeded (p. 63)

A list of resources that were successfully removed from the group by this operation.

Type: Array of strings

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

Pattern: `arn:aws(-[a-z]+)*:[a-z0-9\-]*:([a-z]{2}(-[a-z]+)+-\d{1})?:([0-9]{12})?:.+`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 111).

BadRequestException

The request includes one or more parameters that violate validation rules.

HTTP Status Code: 400

ForbiddenException

The caller isn't authorized to make the request. Check permissions.

HTTP Status Code: 403

InternalServerErrorException

An internal error occurred while processing the request. Try again later.

HTTP Status Code: 500

MethodNotAllowedException

The request uses an HTTP method that isn't allowed for the specified resource.

HTTP Status Code: 405

NotFoundException

One or more of the specified resources don't exist.

HTTP Status Code: 404

TooManyRequestsException

You've exceeded throttling limits by making too many requests in a period of time.

HTTP Status Code: 429

Examples

Example

The following example removes two resources from the specified group.

Sample Request

```
POST /ungroup-resources HTTP/1.1
Host: resource-groups.us-west-2.amazonaws.com
Accept-Encoding: identity
User-Agent: aws-cli/2.2.40 Python/3.8.8 Windows/10 exe/AMD64 prompt/off command/resource-
groups.ungroup-resources
X-Amz-Date: 20220114T205720Z
X-Amz-Security-Token: <SECURITY-TOKEN>
Authorization: AWS4-HMAC-SHA256 Credential=<ACCESS-KEY>/20220113/us-west-2/resource-groups/
aws4_request,SignedHeaders=host;x-amz-date;x-amz-security-token,Signature=<SIGV4-SIGNATURE>
Content-Length: 199

{
    "Group": "CRPGroup",
    "ResourceArns": [
        "arn:aws:ec2:us-west-2:123456789012:capacity-reservation/cr-0070b00d13EXAMPLE",
        "arn:aws:ec2:us-west-2:123456789012:capacity-reservation/cr-061abec820EXAMPLE"
    ]
}
```

Sample Response

```
HTTP/1.1 200 OK
```

```
Date: Fri, 14 Jan 2022 20:57:20 GMT
Content-Type: application/json
Content-Length: 198
x-amzn-RequestId: <VARIES>
x-amz-apigw-id: <VARIES>
X-Amzn-Trace-Id: Root=<VARIES>
Connection: keep-alive

{
    "Succeeded":[
        "arn:aws:ec2:us-west-2:123456789012:capacity-reservation/cr-0070b00d13EXAMPLE",
        "arn:aws:ec2:us-west-2:123456789012:capacity-reservation/cr-061abec820EXAMPLE"
    ],
    "Failed":[],
    "Pending":[]
}
```

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

Untag

Deletes tags from a specified resource group.

Minimum permissions

To run this command, you must have the following permissions:

- `resource-groups:Untag`

Request Syntax

```
PATCH /resources/Arn/tags HTTP/1.1
Content-type: application/json

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

URI Request Parameters

The request uses the following URI parameters.

Arn (p. 66)

The ARN of the resource group from which to remove tags. The command removed both the specified keys and any values associated with those keys.

Length Constraints: Minimum length of 12. Maximum length of 1600.

Pattern: `arn:aws(-[a-z]+)*:resource-groups:[a-z]{2}(-[a-z]+)+-\d{1}:[0-9]{12}:group/[a-zA-Z0-9_\.-]{1,128}`

Required: Yes

Request Body

The request accepts the following data in JSON format.

Keys (p. 66)

The keys of the tags to be removed.

Type: Array of strings

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

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

Required: Yes

Response Syntax

```
HTTP/1.1 200
```

```
Content-type: application/json

{
   "Arn": "string",
   "Keys": [ "string" ]
}
```

Response Elements

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

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

Arn (p. 66)

The ARN of the resource group from which tags have been removed.

Type: String

Length Constraints: Minimum length of 12. Maximum length of 1600.

Pattern: `arn:aws(-[a-z]+)*:resource-groups:[a-z]{2}(-[a-z]+)+-\d{1}:[0-9]{12}:group/[a-zA-Z0-9_\.-]{1,128}`

Keys (p. 66)

The keys of the tags that were removed.

Type: Array of strings

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

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 111).

BadRequestException

The request includes one or more parameters that violate validation rules.

HTTP Status Code: 400

ForbiddenException

The caller isn't authorized to make the request. Check permissions.

HTTP Status Code: 403

InternalServerErrorException

An internal error occurred while processing the request. Try again later.

HTTP Status Code: 500

MethodNotAllowedException

The request uses an HTTP method that isn't allowed for the specified resource.

HTTP Status Code: 405

NotFoundException

One or more of the specified resources don't exist.

HTTP Status Code: 404

TooManyRequestsException

You've exceeded throttling limits by making too many requests in a period of time.

HTTP Status Code: 429

Examples

Example

The following example removes the tag with the key Department from the specified resource group. Note that this updates the tags on only the group itself; it doesn't touch the tags on any group members.

The ARN in the PATCH element must be URL encoded.

Sample Request

```
PATCH /resources/arn%3Aaws%3Aresource-groups%3Aus-west-2%3A123456789012%3Agroup%2FMyTest/
tags HTTP/1.1
Host: resource-groups.us-west-2.amazonaws.com
Accept-Encoding: identity
User-Agent: aws-cli/2.2.40 Python/3.8.8 Windows/10 exe/AMD64 prompt/off command/resource-
groups.untag
X-Amz-Date: 20220120T221301Z
X-Amz-Security-Token: <SECURITY-TOKEN>
Authorization: AWS4-HMAC-SHA256 Credential=<ACCESS-KEY>/20220113/us-west-2/resource-groups/
aws4_request,SignedHeaders=host;x-amz-date;x-amz-security-token,Signature=<SIGV4-SIGNATURE>
Content-Length: 24

{
    "Keys": [
        "Department"
    ]
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: Thu, 20 Jan 2022 22:13:01 GMT
Content-Type: application/json
Content-Length: 91
x-amzn-RequestId: <VARIES>
x-amz-apigw-id: <VARIES>
X-Amzn-Trace-Id: Root=<VARIES>
Connection: keep-alive

{
    "Arn":"arn:aws:resource-groups:us-west-2:123456789012:group/MyTest",
    "Keys":[
        "Department"
    ]
}
```

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

UpdateGroup

Updates the description for an existing group. You cannot update the name of a resource group.

Minimum permissions

To run this command, you must have the following permissions:

- `resource-groups:UpdateGroup`

Request Syntax

```
POST /update-group HTTP/1.1
Content-type: application/json

{
   "Description": "string",
   "Group": "string",
   "GroupName": "string"
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

Description (p. 70)

The new description that you want to update the resource group with. Descriptions can contain letters, numbers, hyphens, underscores, periods, and spaces.

Type: String

Length Constraints: Maximum length of 512.

Pattern: `[\sa-zA-Z0-9_\.-]*`

Required: No

Group (p. 70)

The name or the ARN of the resource group to modify.

Type: String

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

Pattern: `(arn:aws(-[a-z]+)*:resource-groups:[a-z]{2}(-[a-z]+)+-\d{1}:[0-9]{12}:group/)?[a-zA-Z0-9_\.-]{1,128}`

Required: No

GroupName (p. 70)

This parameter has been deprecated.

Don't use this parameter. Use `Group` instead.

Type: String

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

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

Required: No

Response Syntax

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

{
   "Group": {
      "Description": "string",
      "GroupArn": "string",
      "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.

Group (p. 71)

The update description of the resource group.

Type: Group (p. 80) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 111).

BadRequestException

The request includes one or more parameters that violate validation rules.

HTTP Status Code: 400

ForbiddenException

The caller isn't authorized to make the request. Check permissions.

HTTP Status Code: 403

InternalServerErrorException

An internal error occurred while processing the request. Try again later.

HTTP Status Code: 500

MethodNotAllowedException

The request uses an HTTP method that isn't allowed for the specified resource.

HTTP Status Code: 405

NotFoundException

One or more of the specified resources don't exist.

HTTP Status Code: 404

TooManyRequestsException

You've exceeded throttling limits by making too many requests in a period of time.

HTTP Status Code: 429

Examples

Example

The following example adds or updates the description attached to the specified resource group.

Sample Request

```
POST /update-group HTTP/1.1
Host: resource-groups.us-west-2.amazonaws.com
Accept-Encoding: identity
User-Agent: aws-cli/2.2.40 Python/3.8.8 Windows/10 exe/AMD64 prompt/off command/resource-
groups.update-group
X-Amz-Date: 20220120T224252Z
X-Amz-Security-Token: <SECURITY-TOKEN>
Authorization: AWS4-HMAC-SHA256 Credential=<ACCESS-KEY>/20220113/us-west-2/resource-groups/
aws4_request,SignedHeaders=host;x-amz-date;x-amz-security-token,Signature=<SIGV4-SIGNATURE>
Content-Length: 51

{
    "Group": "MyTest",
    "Description": "My Test group"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: Thu, 20 Jan 2022 22:42:52 GMT
Content-Type: application/json
Content-Length: 155
x-amzn-RequestId: <VARIES>
x-amz-apigw-id: <VARIES>
X-Amzn-Trace-Id: Root=<VARIES>
Connection: keep-alive

{
    "Group":{
        "GroupArn":"arn:aws:resource-groups:us-west-2:123456789012:group/MyTest",
        "Name":"MyTest",
        "Description":"My Test group",
        "OwnerId":"123456789012"
    }
```

```
}
```

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

UpdateGroupQuery

Updates the resource query of a group. For more information about resource queries, see Create a tag-based group in Resource Groups.

Minimum permissions

To run this command, you must have the following permissions:

- `resource-groups:UpdateGroupQuery`

Request Syntax

```
POST /update-group-query HTTP/1.1
Content-type: application/json

{
   "Group": "string",
   "GroupName": "string",
   "ResourceQuery": {
      "Query": "string",
      "Type": "string"
   }
}
```

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request accepts the following data in JSON format.

ResourceQuery (p. 74)

The resource query to determine which AWS resources are members of this resource group.

Note
A resource group can contain either a `Configuration` or a `ResourceQuery`, but not both.

Type: ResourceQuery (p. 94) object

Required: Yes

Group (p. 74)

The name or the ARN of the resource group to query.

Type: String

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

Pattern: `(arn:aws(-[a-z]+)*:resource-groups:[a-z]{2}(-[a-z]+)+-\d{1}:[0-9]{12}:group/)?[a-zA-Z0-9_\.-]{1,128}`

Required: No

GroupName (p. 74)

This parameter has been deprecated.

Don't use this parameter. Use `Group` instead.

Type: String

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

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

Required: No

Response Syntax

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

{
   "GroupQuery": {
      "GroupName": "string",
      "ResourceQuery": {
         "Query": "string",
         "Type": "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.

GroupQuery (p. 75)

The updated resource query associated with the resource group after the update.

Type: GroupQuery (p. 88) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 111).

BadRequestException

The request includes one or more parameters that violate validation rules.

HTTP Status Code: 400

ForbiddenException

The caller isn't authorized to make the request. Check permissions.

HTTP Status Code: 403

InternalServerErrorException

An internal error occurred while processing the request. Try again later.

HTTP Status Code: 500

MethodNotAllowedException

The request uses an HTTP method that isn't allowed for the specified resource.

HTTP Status Code: 405

NotFoundException

One or more of the specified resources don't exist.

HTTP Status Code: 404

TooManyRequestsException

You've exceeded throttling limits by making too many requests in a period of time.

HTTP Status Code: 429

Examples

Example

The following example updates the query of the specified resource group. A resource group query requires that you supply both a `Type` parameter and a `Query` string parameter. The `Query` parameter is a JSON representation of the query as a single string.

Sample Request

```
POST /update-group-query HTTP/1.1
Host: resource-groups.us-west-2.amazonaws.com
Accept-Encoding: identity
User-Agent: aws-cli/2.2.40 Python/3.8.8 Windows/10 exe/AMD64 prompt/off command/resource-
groups.update-group-query
X-Amz-Date: 20220120T225422Z
X-Amz-Security-Token: <SECURITY-TOKEN>
Authorization: AWS4-HMAC-SHA256 Credential=<ACCESS-KEY>/20220113/us-west-2/resource-groups/
aws4_request,SignedHeaders=host;x-amz-date;x-amz-security-token,Signature=<SIGV4-SIGNATURE>
Content-Length: 197

{
    "GroupName": "MyTest",
    "ResourceQuery": {
        "Type": "TAG_FILTERS_1_0",
        "Query": "{\"ResourceTypeFilters\":[\"AWS::EC2::Instance\"],\"TagFilters\":[{\"Key
\":\"Name\", \"Values\":[\"WebServers\"]}]}"}}
```

Sample Response

```
HTTP/1.1 200 OK
Date: Thu, 20 Jan 2022 22:54:23 GMT
Content-Type: application/json
Content-Length: 206
x-amzn-RequestId: <VARIES>
x-amz-apigw-id: <VARIES>
X-Amzn-Trace-Id: Root=<VARIES>
Connection: keep-alive

{
    "GroupQuery":{
```

```
        "GroupName":"MyTest",
        "ResourceQuery":{
            "Type":"TAG_FILTERS_1_0",
            "Query":"{\"ResourceTypeFilters\":[\"AWS::EC2::Instance\"],\"TagFilters\":
[{\"Key\":\"Name\", \"Values\":[\"WebServers\"]}]}"
        }
    }
}
```

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 Resource Groups 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:

FailedResource

A resource that failed to be added to or removed from a group.

Contents

Note
In the following list, the required parameters are described first.

ErrorCode

The error code associated with the failure.

Type: String

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

Required: No

ErrorMessage

The error message text associated with the failure.

Type: String

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

Required: No

ResourceArn

The ARN of the resource that failed to be added or removed.

Type: String

Pattern: `arn:aws(-[a-z]+)*:[a-z0-9\-]*:([a-z]{2}(-[a-z]+)+-\d{1})?:([0-9]{12})?:.+`

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

Group

A resource group that contains AWS resources. You can assign resources to the group by associating either of the following elements with the group:

- ResourceQuery (p. 94) - Use a resource query to specify a set of tag keys and values. All resources in the same AWS Region and AWS account that have those keys with the same values are included in the group. You can add a resource query when you create the group, or later by using the PutGroupConfiguration (p. 48) operation.
- GroupConfiguration (p. 82) - Use a service configuration to associate the group with an AWS service. The configuration specifies which resource types can be included in the group.

Contents

Note
In the following list, the required parameters are described first.

GroupArn

The ARN of the resource group.

Type: String

Length Constraints: Minimum length of 12. Maximum length of 1600.

Pattern: `arn:aws(-[a-z]+)*:resource-groups:[a-z]{2}(-[a-z]+)+-\d{1}:[0-9]{12}:group/[a-zA-Z0-9_\.-]{1,128}`

Required: Yes

Name

The name of the resource group.

Type: String

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

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

Required: Yes

Description

The description of the resource group.

Type: String

Length Constraints: Maximum length of 512.

Pattern: `[\sa-zA-Z0-9_\.-]*`

Required: No

See Also

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

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

GroupConfiguration

A service configuration associated with a resource group. The configuration options are determined by the AWS service that defines the `Type`, and specifies which resources can be included in the group. You can add a service configuration when you create the group by using CreateGroup (p. 3), or later by using the PutGroupConfiguration (p. 48) operation. For details about group service configuration syntax, see Service configurations for resource groups.

Contents

Note
In the following list, the required parameters are described first.

Configuration

The configuration currently associated with the group and in effect.

Type: Array of GroupConfigurationItem (p. 84) objects

Array Members: Maximum number of 2 items.

Required: No

FailureReason

If present, the reason why a request to update the group configuration failed.

Type: String

Required: No

ProposedConfiguration

If present, the new configuration that is in the process of being applied to the group.

Type: Array of GroupConfigurationItem (p. 84) objects

Array Members: Maximum number of 2 items.

Required: No

Status

The current status of an attempt to update the group configuration.

Type: String

Valid Values: `UPDATING | UPDATE_COMPLETE | UPDATE_FAILED`

Required: No

See Also

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

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

GroupConfigurationItem

An item in a group configuration. A group service configuration can have one or more items. For details about group service configuration syntax, see Service configurations for resource groups.

Contents

Note
In the following list, the required parameters are described first.

Type

Specifies the type of group configuration item. Each item must have a unique value for `type`. For the list of types that you can specify for a configuration item, see Supported resource types and parameters.

Type: String

Length Constraints: Maximum length of 40.

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

Required: Yes

Parameters

A collection of parameters for this group configuration item. For the list of parameters that you can use with each configuration item type, see Supported resource types and parameters.

Type: Array of GroupConfigurationParameter (p. 85) objects

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

GroupConfigurationParameter

A parameter for a group configuration item. For details about group service configuration syntax, see Service configurations for resource groups.

Contents

Note
In the following list, the required parameters are described first.

Name

The name of the group configuration parameter. For the list of parameters that you can use with each configuration item type, see Supported resource types and parameters.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 80.

Pattern: `[a-z-]+`

Required: Yes

Values

The value or values to be used for the specified parameter. For the list of values you can use with each parameter, see Supported resource types and parameters.

Type: Array of strings

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

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

Required: No

See Also

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

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

GroupFilter

A filter collection that you can use to restrict the results from a `List` operation to only those you want to include.

Contents

Note
In the following list, the required parameters are described first.

Name

The name of the filter. Filter names are case-sensitive.

Type: String

Valid Values: `resource-type | configuration-type`

Required: Yes

Values

One or more filter values. Allowed filter values vary by group filter name, and are case-sensitive.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 5 items.

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

Pattern: `AWS::(AllSupported|[a-zA-Z0-9]+::[a-zA-Z0-9]+)`

Required: Yes

See Also

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

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

GroupIdentifier

The unique identifiers for a resource group.

Contents

Note
In the following list, the required parameters are described first.

GroupArn

The ARN of the resource group.

Type: String

Length Constraints: Minimum length of 12. Maximum length of 1600.

Pattern: `arn:aws(-[a-z]+)*:resource-groups:[a-z]{2}(-[a-z]+)+-\d{1}:[0-9]{12}:group/[a-zA-Z0-9_\.-]{1,128}`

Required: No

GroupName

The name of the resource group.

Type: String

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

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

Required: No

See Also

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

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

GroupQuery

A mapping of a query attached to a resource group that determines the AWS resources that are members of the group.

Contents

Note
In the following list, the required parameters are described first.

GroupName

The name of the resource group that is associated with the specified resource query.

Type: String

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

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

Required: Yes

ResourceQuery

The resource query that determines which AWS resources are members of the associated resource group.

Type: ResourceQuery (p. 94) object

Required: Yes

See Also

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

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

ListGroupResourcesItem

A structure returned by the ListGroupResources (p. 37) operation that contains identity and group membership status information for one of the resources in the group.

Contents

Note
In the following list, the required parameters are described first.

Identifier

A structure that contains the ARN of a resource and its resource type.

Type: ResourceIdentifier (p. 93) object

Required: No

Status

A structure that contains the status of this resource's membership in the group.

Note
This field is present in the response only if the group is of type `AWS::EC2::HostManagement`.

Type: ResourceStatus (p. 97) object

Required: No

See Also

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

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

PendingResource

A structure that identifies a resource that is currently pending addition to the group as a member. Adding a resource to a resource group happens asynchronously as a background task and this one isn't completed yet.

Contents

Note
In the following list, the required parameters are described first.

ResourceArn

The Amazon resource name (ARN) of the resource that's in a pending state.

Type: String

Pattern: `arn:aws(-[a-z]+)*:[a-z0-9\-]*:([a-z]{2}(-[a-z]+)+-\d{1})?:([0-9]{12})?:.+`

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

QueryError

A two-part error structure that can occur in `ListGroupResources` or `SearchResources` operations on CloudFront stack-based queries. The error occurs if the CloudFront stack on which the query is based either does not exist, or has a status that renders the stack inactive. A `QueryError` occurrence does not necessarily mean that AWS Resource Groups could not complete the operation, but the resulting group might have no member resources.

Contents

Note
In the following list, the required parameters are described first.

ErrorCode

Possible values are `CLOUDFORMATION_STACK_INACTIVE` and `CLOUDFORMATION_STACK_NOT_EXISTING`.

Type: String

Valid Values: `CLOUDFORMATION_STACK_INACTIVE | CLOUDFORMATION_STACK_NOT_EXISTING`

Required: No

Message

A message that explains the `ErrorCode` value. Messages might state that the specified CloudFront stack does not exist (or no longer exists). For `CLOUDFORMATION_STACK_INACTIVE`, the message typically states that the CloudFront stack has a status that is not (or no longer) active, such as `CREATE_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

ResourceFilter

A filter name and value pair that is used to obtain more specific results from a list of resources.

Contents

Note
In the following list, the required parameters are described first.

Name

The name of the filter. Filter names are case-sensitive.

Type: String

Valid Values: `resource-type`

Required: Yes

Values

One or more filter values. Allowed filter values vary by resource filter name, and are case-sensitive.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 5 items.

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

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

Required: Yes

See Also

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

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

ResourceIdentifier

A structure that contains the ARN of a resource and its resource type.

Contents

Note
In the following list, the required parameters are described first.

ResourceArn

The ARN of a resource.

Type: String

Pattern: `arn:aws(-[a-z]+)*:[a-z0-9\-]*:([a-z]{2}(-[a-z]+)+-\d{1})?:([0-9]{12})?:.+`

Required: No

ResourceType

The resource type of a resource, such as `AWS::EC2::Instance`.

Type: String

Pattern: `AWS::[a-zA-Z0-9]+::\w+`

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

ResourceQuery

The query you can use to define a resource group or a search for resources. A `ResourceQuery` specifies both a query `Type` and a `Query` string as JSON string objects. See the examples section for example JSON strings. For more information about creating a resource group with a resource query, see Build queries and groups in AWS Resource Groups in the *AWS Resource Groups User Guide*

When you combine all of the elements together into a single string, any double quotes that are embedded inside another double quote pair must be escaped by preceding the embedded double quote with a backslash character (\). For example, a complete `ResourceQuery` parameter must be formatted like the following CLI parameter example:

```
--resource-query '{"Type":"TAG_FILTERS_1_0","Query":"{\"ResourceTypeFilters\":
[\"AWS::AllSupported\"],\"TagFilters\":[{\"Key\":\"Stage\",\"Values\":[\"Test
\"]}]}"}'
```

In the preceding example, all of the double quote characters in the value part of the `Query` element must be escaped because the value itself is surrounded by double quotes. For more information, see Quoting strings in the *AWS Command Line Interface User Guide*.

For the complete list of resource types that you can use in the array value for `ResourceTypeFilters`, see Resources you can use with AWS Resource Groups and Tag Editor in the *AWS Resource Groups User Guide*. For example:

```
"ResourceTypeFilters":["AWS::S3::Bucket", "AWS::EC2::Instance"]
```

Contents

Note
In the following list, the required parameters are described first.

Query

The query that defines a group or a search. The contents depends on the value of the `Type` element.

- `ResourceTypeFilters` – Applies to all `ResourceQuery` objects of either `Type`. This element contains one of the following two items:
  - The value `AWS::AllSupported`. This causes the ResourceQuery to match resources of any resource type that also match the query.
  - A list (a JSON array) of resource type identifiers that limit the query to only resources of the specified types. For the complete list of resource types that you can use in the array value for `ResourceTypeFilters`, see Resources you can use with AWS Resource Groups and Tag Editor in the *AWS Resource Groups User Guide*.

  Example: `"ResourceTypeFilters": ["AWS::AllSupported"]` or `"ResourceTypeFilters": ["AWS::EC2::Instance", "AWS::S3::Bucket"]`
- `TagFilters` – applicable only if `Type` = `TAG_FILTERS_1_0`. The `Query` contains a JSON string that represents a collection of simple tag filters. The JSON string uses a syntax similar to the `GetResources` operation, but uses only the `ResourceTypeFilters` and `TagFilters` fields. If you specify more than one tag key, only resources that match all tag keys, and at least one value of each specified tag key, are returned in your query. If you specify more than one value for a tag key, a resource matches the filter if it has a tag key value that matches *any* of the specified values.

  For example, consider the following sample query for resources that have two tags, `Stage` and `Version`, with two values each:

  ```
  [{"Stage":["Test","Deploy"]},{"Version":["1","2"]}]
  ```

The results of this resource query could include the following.

- An Amazon EC2 instance that has the following two tags: `{"Stage":"Deploy"}`, and `{"Version":"2"}`
- An S3 bucket that has the following two tags: `{"Stage":"Test"}`, and `{"Version":"1"}`

The resource query results would *not* include the following items in the results, however.

- An Amazon EC2 instance that has only the following tag: `{"Stage":"Deploy"}`.

  The instance does not have **all** of the tag keys specified in the filter, so it is excluded from the results.
- An RDS database that has the following two tags: `{"Stage":"Archived"}` and `{"Version":"4"}`

  The database has all of the tag keys, but none of those keys has an associated value that matches at least one of the specified values in the filter.

Example: `"TagFilters": [ { "Key": "Stage", "Values": [ "Gamma", "Beta" ] }`

- `StackIdentifier` – applicable only if `Type` = `CLOUDFORMATION_STACK_1_0`. The value of this parameter is the Amazon Resource Name (ARN) of the AWS CloudFormation stack whose resources you want included in the group.

Type: String

Length Constraints: Maximum length of 4096.

Pattern: `[\s\S]*`

Required: Yes

Type

The type of the query to perform. This can have one of two values:

- *`CLOUDFORMATION_STACK_1_0:`* Specifies that you want the group to contain the members of an AWS CloudFormation stack. The `Query` contains a `StackIdentifier` element with an ARN for a CloudFormation stack.
- *`TAG_FILTERS_1_0:`* Specifies that you want the group to include resource that have tags that match the query.

Type: String

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

Pattern: `^\w+$`

Valid Values: `TAG_FILTERS_1_0 | CLOUDFORMATION_STACK_1_0`

Required: Yes

See Also

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

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

ResourceStatus

A structure that identifies the current group membership status for a resource. Adding a resource to a resource group is performed asynchronously as a background task. A `PENDING` status indicates, for this resource, that the process isn't completed yet.

Contents

Note
In the following list, the required parameters are described first.

Name

The current status.

Type: String

Valid Values: `PENDING`

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

Service configurations for resource groups

Resource groups enable you to manage collections of your AWS resources as a unit. Some AWS services support this by performing requested operations on all members of the group. Such services can store the settings to be applied to group members as a *configuration* in the form of a JSON data structure that is attached to the group.

This topic describes the available configuration settings for supported AWS services.

Topics

How to access the service configuration attached to a resource group

Services that support service-linked groups typically set the configuration for you when you use the tools provided by that service, such as that service's management console or its AWS CLI and AWS SDK operations. Some services fully manage their service-linked groups and you can't modify them in any way except as allowed by the console or commands provided by the owning AWS service. However, in some cases, you can interact with the service configuration by using the following API operations in the AWS SDKs or their AWS CLI equivalents:

- You can attach your own configuration to a group when you create the group by using the `CreateGroup` operation.
- You can modify the current configuration attached to a group by using the `PutGroupConfiguration` operation.
- You can view the current configuration of a resource group by calling the `GetGroupConfiguration` operation.

JSON syntax of a service configuration

A resource group can contain a *configuration* that defines service-specific settings that apply to the resources that are members of that group.

A configuration is expressed as a JSON object. At the top-most level, a configuration is an array of group configuration items. Each group configuration item contains two elements: a `Type` for the configuration and a set of `Parameters` defined by that type. Each parameter contains a `Name` and an array of one or more `Values`. The following example with *`placeholders`* shows the basic syntax for a configuration for a single sample resource type. This example shows a type with two parameters, and each parameter with two values. The actual valid types, parameters, and values are discussed in the next section.

```
{
    "Configuration": [
        {
            "Type": "configuration-type",
```

```
            "Parameters": [
                {
                    "Name": "parameter1-name",
                    "Values": [
                        "value1",
                        "value2"
                    ]
                },
                {
                    "Name": "parameter2-name",
                    "Values": [
                        "value3",
                        "value4"
                    ]
                }
            ]
        }
    ]
}
```

Supported configuration types and parameters

Resource Groups supports using the following configuration types. Each configuration type has a set of parameters that are valid for that type.

Topics

AWS::ResourceGroups::Generic

This configuration type specifies settings that enforce membership requirements on the resource group, rather than configuring the behavior of a specific resource type for an AWS service. This configuration type is automatically added by those service-linked groups that need it, such as the `AWS::EC2::CapacityReservationPool` and `AWS::EC2::HostManagment` types.

The following `Parameters` are valid for the `AWS::ResourceGroups::Generic` service-linked group `Type`.

- **`allowed-resource-types`**

  This parameter specifies that the resource group can consist of resources of only the specified type or types.

  Data type of values: String

  Permitted values:

  - `AWS::EC2::Host` – A `Configuration` with this parameter and value is required when the service configuration also contains a `Configuration` of type `AWS::EC2::HostManagement`. This ensures that the `HostManagement` group can contain only Amazon EC2 dedicated hosts.
  - `AWS::EC2::CapacityReservation` – A `Configuration` with this parameter and value is required when the service configuration also contains a `Configuration` item of type

`AWS::EC2::CapacityReservationPool`. This ensures that a `CapacityReservation` group can contain only Amazon EC2 capacity reservation capacity.

Required: Conditional, based on other `Configuration` elements that are attached to the resource group. See the previous entry for **Permitted values**.

The following example restricts group members to only Amazon EC2 host instances.

```
{
    "Configuration": [
        {
            "Type": "AWS::ResourceGroups::Generic",
            "Parameters": [
                {
                    "Name": "allowed-resource-types",
                    "Values": ["AWS::EC2::Host"]
                }
            ]
        }
    ]
}
```

- **`deletion-protection`**

  This parameter specifies that the resource group can't be deleted unless it contains no members. For more information, see Delete a host resource group in the *License Manager User Guide*

  Data type of values: Array of string

  Permitted values: The only permitted value is `[ "UNLESS_EMPTY" ]` (the value must be upper case).

  Required: Conditional, based on other `Configuration` elements that are attached to the resource group. This parameter is required only when the resource group also has another `Configuration` element with the `Type` of `AWS::EC2::HostManagement`.

  The following example enables delete protection for the group unless the group has no members.

```
{
    "Configuration": [
        {
            "Type": "AWS::ResourceGroups::Generic",
            "Parameters": [
                {
                    "Name": "deletion-protection",
                    "Values": [ "UNLESS_EMPTY" ]
                }
            ]
        }
    ]
}
```

AWS::AppRegistry::Application

This `Configuration` type specifies that the resource group represents an application created by AWS Service Catalog AppRegistry.

Resource groups of this type are fully managed by the AWS Service Catalog AppRegistry service, and can't be created, updated, or deleted by users other than by using the tools provided by AWS Service Catalog AppRegistry.

> **Note**
> Because resource groups of this type are automatically created and maintained by AWS and not managed by the user, these resource groups do not count against your quota limit for the maximum number of resource groups that you can create in your AWS account.

For more information, see Using AWS Service Catalog AppRegistry in the *AWS Service Catalog User Guide*.

When AWS Service Catalog AppRegistry creates a service-linked resource group of this type, it also automatically creates a separate, additional AWS CloudFormation service-linked group (p. 102) for each AWS CloudFormation stack associated with the application.

AWS Service Catalog AppRegistry automatically names the service-linked groups of this type that its creates with the prefix `AWS_AppRegistry_Application-` followed by the name of the application: `AWS_AppRegistry_Application-`*`MyAppName`*

The following parameters are supported for the `AWS::AppRegistry::Application` service-linked group type.

- **`Name`**

  This parameter specifies the friendly name of the application that was assigned by the user when it was created in AWS Service Catalog AppRegistry.

  Data type of values: String

  Permitted values: any text string permitted by the AWS Service Catalog AppRegistry service for an application name.

  Required: Yes

- **`Arn`**

  This parameter specifies the Amazon Resource Name (ARN) path of the application assigned by AWS Service Catalog AppRegistry.

  Data type of values: String

  Permitted values: a valid ARN.

  Required: Yes

> **Note**
> To change any of these elements, you must modify the application using the AWS Service Catalog AppRegistry console or that service's AWS SDK and AWS CLI operations.

This application resource group automatically includes as group members the resource groups created for the AWS CloudFormation stacks (p. 102) that are associated with the AWS Service Catalog AppRegistry application. You can use the ListGroupResources operation to see those child groups.

The following example shows what the configuration section of a `AWS::AppRegistry::Application` service-linked group looks like.

```
{
    "Configuration": [
        {
            "Type": "AWS::AppRegistry::Application",
            "Parameters":[
                {
                    "Name": "Name",
                    "Values": [
```

```
                        "MyApplication"
                    ]
                },
                {
                    "Name": "Arn",
                    "Values": [
                        "arn:aws:servicecatalog:us-east-1:123456789012:/
applications/<application-id>"
                    ]
                }
            ]
        }
    ]
}
```

AWS::CloudFormation::Stack

This `Configuration` type specifies that the group represents an AWS CloudFormation stack and its members are the AWS resources created by that stack.

Resource groups of this type are automatically created for you when you associate a AWS CloudFormation stack with the AWS Service Catalog AppRegistry service. You can't create, update, or delete these groups except by using the tools provided by AWS Service Catalog AppRegistry.

AWS Service Catalog AppRegistry automatically names the service-linked groups of this type that its creates with the prefix `AWS_CloudFormation_Stack-` followed by the name of the stack: `AWS_CloudFormation_Stack-MyStackName`

> **Note**
> Because resource groups of this type are automatically created and maintained by AWS and not managed by the user, these resource groups do not count against your quota limit for the maximum number of resource groups that you can create in your AWS account.

For more information, see Using AWS Service Catalog AppRegistry in the *AWS Service Catalog User Guide*.

AWS Service Catalog AppRegistry automatically creates a service-linked resource group of this type for every AWS CloudFormation stack that you associate with the AWS Service Catalog AppRegistry application. These resource groups become child members of the parent resource group for the AWS Service Catalog AppRegistry application (p. 100).

The members of this AWS CloudFormation resource group are the AWS resources created as part of the stack.

The following parameters are supported for the `AWS::CloudFormation::Stack` service-linked group type.

- **`Name`**

  This parameter specifies the friendly name of the AWS CloudFormation stack assigned by the user when the stack was created.

  Data type of values: String

  Permitted values: any text string permitted by the AWS CloudFormation service for a stack name.

  Required: Yes

- **`Arn`**

  This parameter specifies the Amazon Resource Name (ARN) path of the AWS CloudFormation stack attached to the application in AWS Service Catalog AppRegistry.

Data type of values: String

Permitted values: a valid ARN.

Required: Yes

Note
To change any of these elements, you must modify the application using the AWS Service Catalog AppRegistry console or equivalent AWS SDK and AWS CLI operations.

The following example shows what the configuration section of an `AWS::CloudFormation::Stack` service-linked group looks like.

```
{
    "Configuration": [
        {
            "Type": "AWS::CloudFormation::Stack",
            "Parameters":[
                {
                    "Name": "Name",
                    "Values": [
                        "MyStack"
                    ]
                },
                {
                    "Name": "Arn",
                    "Values": [
                        "arn:aws:cloudformation:us-
east-1:123456789012:stack/MyStack/<stack-id>"
                    ]
                }
            ]
        }
    ]
}
```

AWS::EC2::CapacityReservationPool

This `Configuration` type specifies that the resource group represents a common pool of capacity provided by the group's members. The members of this resource group are required to be Amazon EC2 capacity reservations. This lets you launch an Amazon EC2 instance using this group as the value for the capacity reservation parameter. When you do this, the instance uses the available reserved capacity in the group. If resource group has no available capacity, the instance launches as a stand alone on-demand instance outside of the pool. For more information, see Working with Capacity Reservation groups in the *Amazon EC2 User Guide for Linux Instances*.

If you configure a service-linked resource group with a `Configuration` item of this type, then you must also specify separate `Configuration` items with the following values:

- An `AWS::ResourceGroups::Generic` type with one parameter:
  - The parameter `allowed-resource-types` and a single value of `AWS::EC2::CapacityReservation`. This ensures that only Amazon EC2 capacity reservations can be members of the resource group.

The `AWS::EC2::CapacityReservationPool` item in a group configuration doesn't support any parameters.

The following example shows what the `Configuration` section of such a group looks like.

```
{
    "Configuration": [
        {
            "Type": "AWS::EC2::CapacityReservationPool"
        },
        {
            "Type": "AWS::ResourceGroups::Generic",
            "Parameters": [
                {
                    "Name": "allowed-resource-types",
                    "Values": [ "AWS::EC2::CapacityReservation" ]
                }
            ]
        }
    ]
}
```

AWS::EC2::HostManagement

This identifier specifies settings for Amazon EC2 host management and AWS License Manager that are enforced for the group's members. For more information, see Host resource groups in AWS License Manager.

If you configure a service-linked resource group with a `Configuration` item of this type, then you must also specify separate `Configuration` items with the following values:

- An `AWS::ResourceGroups::Generic` type, with a parameter of `allowed-resource-types` and a single value of `AWS::EC2::Host`. This ensures that only Amazon EC2 dedicated hosts can be members of the group.
- An `AWS::ResourceGroups::Generic` type, with a parameter of `deletion-protection` and a single value of `UNLESS_EMPTY`. This ensures that the group can't be deleted unless the group is empty.

The following parameters are supported for the `AWS::EC2::HostManagement` service-linked group type.

- **`auto-allocate-host`**

  This parameter specifies whether instances are launched onto a specific dedicated host, or onto any available host that has a matching configuration. For more information, see Understanding auto-placement and affinity in the *Amazon EC2 User Guide for Linux Instances*.

  Data type of values: Boolean

  Permitted values: "true" or "false" (must be lower case).

  Required: No

```
{
    "Configuration": [
        {
            "Type": "AWS::EC2::HostManagement",
            "Parameters": [
                {
                    "Name": "auto-allocate-host",
                    "Values": [ "true" ]
                }
            ]
        },
        {
```

```
            "Type": "AWS::ResourceGroups::Generic",
            "Parameters": [
                {
                    "Name": "allowed-resource-types",
                    "Values": [ "AWS::EC2::Host" ]
                },
                {
                    "Name": "deletion-protection",
                    "Values": [ "UNLESS_EMPTY" ]
                }
            ]
        }
    ]
}
```

- **`auto-release-host`**

  This parameter specifies whether a dedicated host in the group is automatically released after its last running instance is terminated. For more information, see Releasing Dedicated Hosts in the *Amazon EC2 User Guide for Linux Instances*.

  Data type of values: Boolean

  Permitted values: "true" or "false" (must be lower case).

  Required: No

```
{
    "Configuration": [
        {
            "Type": "AWS::EC2::HostManagement",
            "Parameters": [
                {
                    "Name": "auto-release-host",
                    "Values": [ "false" ]
                }
            ]
        },
        {
            "Type": "AWS::ResourceGroups::Generic",
            "Parameters": [
                {
                    "Name": "allowed-resource-types",
                    "Values": [ "AWS::EC2::Host" ]
                },
                {
                    "Name": "deletion-protection",
                    "Values": [ "UNLESS_EMPTY" ]
                }
            ]
        }
    ]
}
```

- **`allowed-host-families`**

  This parameter specifies which instance type families can be used by instances that are members of this group.

  Data type of values: An array of String.

  Permitted values: Each must be a valid Amazon EC2 instance type family identifier, such as `C4`, `M5`, `P3dn`, or `R5d`.

Required: No

The following example configuration item specifies that launched instances can be only members of the C5 or M5 instance type families.

```
{
    "Configuration": [
        {
            "Type": "AWS::EC2::HostManagement",
            "Parameters": [
                {
                    "Name": "allowed-host-families",
                    "Values": ["c5", "m5"]
                }
            ]
        },
        {
            "Type": "AWS::ResourceGroups::Generic",
            "Parameters": [
                {
                    "Name": "allowed-resource-types",
                    "Values": ["AWS::EC2::Host"]
                },
                {
                    "Name": "deletion-protection",
                    "Values": ["UNLESS_EMPTY"]
                }
            ]
        }
    ]
}
```

- **`allowed-host-based-license-configurations`**

  This parameter specifies the Amazon Resource Name (ARN) paths of one or more core/socket based license configurations that you want applied to members of the group.

  Data type of values: An array of ARNs.

  Permitted values: Each must be a valid License Manager configuration ARN.

  Required: Conditional. You must specify either this parameter or `any-host-based-license-configuration`, but not both. They are mutually exclusive.

  The following example configuration item specifies that group members can use the two specified License Manager configurations.

```
{
    "Configuration": [
        {
            "Type": "AWS::EC2::HostManagement",
            "Parameters": [
                {
                    "Name": "allowed-host-based-license-configurations",
                    "Values": [
                        "arn:aws:license-manager:us-west-2:123456789012:license-
configuration:lic-6eb6586f508a786a2ba41EXAMPLE1111",
                        "arn:aws:license-manager:us-west-2:123456789012:license-
configuration:lic-8a786a26f50ba416eb658EXAMPLE2222"
                    ]
                }
            ]
```

```
        },
        {
            "Type": "AWS::ResourceGroups::Generic",
            "Parameters": [
                {
                    "Name": "allowed-resource-types",
                    "Values": [ "AWS::EC2::Host" ]
                },
                {
                    "Name": "deletion-protection",
                    "Values": [ "UNLESS_EMPTY" ]
                }
            ]
        }
    ]
}
```

- **`any-host-based-license-configuration`**

  This parameter specifies that you do not want to associate a specific license configuration to your group. In this case, all core/socket based license configurations are available to your members of your host resource group. Use this setting if you have an unlimited number of licenses and want to optimize for host utilization.

  Data type of values: Boolean

  Permitted values: "true" or "false" (must be lower case).

  Required: Conditional. You must specify either this parameter or `allowed-host-based-license-configurations`, but not both. They are mutually exclusive.

  The following example configuration item specifies that group members can use any core/socket based license configuration.

```
{
    "Configuration": [
        {
            "Type": "AWS::EC2::HostManagement",
            "Parameters": [
                {
                    "Name": "any-host-based-license-configuration",
                    "Values": ["true"]
                }
            ]
        },
        {
            "Type": "AWS::ResourceGroups::Generic",
            "Parameters": [
                {
                    "Name": "allowed-resource-types",
                    "Values": ["AWS::EC2::Host"]
                },
                {
                    "Name": "deletion-protection",
                    "Values": ["UNLESS_EMPTY"]
                }
            ]
        }
    ]
}
```

The following example illustrates how to include all of the host management settings together in a single configuration.

```
{
    "Configuration": [
        {
            "Type": "AWS::EC2::HostManagement",
            "Parameters": [
                {
                    "Name": "auto-allocate-host",
                    "Values": ["true"]
                },
                {
                    "Name": "auto-release-host",
                    "Values": ["false"]
                },
                {
                    "Name": "allowed-host-families",
                    "Values": ["c5", "m5"]
                },
                {
                    "Name": "allowed-host-based-license-configurations",
                    "Values": [
                        "arn:aws:license-manager:us-west-2:123456789012:license-
configuration:lic-6eb6586f508a786a2ba41EXAMPLE1111",
                        "arn:aws:license-manager:us-west-2:123456789012:license-
configuration:lic-8a786a26f50ba416eb658EXAMPLE2222"
                    ]
                }
            ]
        },
        {
            "Type": "AWS::ResourceGroups::Generic",
            "Parameters": [
                {
                    "Name": "allowed-resource-types",
                    "Values": ["AWS::EC2::Host"]
                },
                {
                    "Name": "deletion-protection",
                    "Values": ["UNLESS_EMPTY"]
                }
            ]
        }
    ]
}
```

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