AWS Application Discovery Service

API Reference

API Version 2015-11-01

AWS Application Discovery Service: API Reference

Table of Contents

Welcome

AWS Application Discovery Service helps you plan application migration projects by automatically identifying servers, virtual machines (VMs), software, and software dependencies running in your on-premises data centers. Application Discovery Service also collects application performance data, which can help you assess the outcome of your migration. The data collected by Application Discovery Service is securely retained in an AWS-hosted and managed database in the cloud. You can export the data as a CSV or XML file into your preferred visualization tool or cloud-migration solution to plan your migration. For more information, see AWS Application Discovery Service FAQ.

Application Discovery Service offers two modes of operation:

- **Agentless discovery** mode is recommended for environments that use VMware vCenter Server. This mode doesn't require you to install an agent on each host. Agentless discovery gathers server information regardless of the operating systems, which minimizes the time required for initial on-premises infrastructure assessment. Agentless discovery doesn't collect information about software and software dependencies. It also doesn't work in non-VMware environments.
- **Agent-based discovery** mode collects a richer set of data than agentless discovery by using the AWS Application Discovery Agent, which you install on one or more hosts in your data center. The agent captures infrastructure and application information, including an inventory of installed software applications, system and process performance, resource utilization, and network dependencies between workloads. The information collected by agents is secured at rest and in transit to the Application Discovery Service database in the cloud.

We recommend that you use agent-based discovery for non-VMware environments and to collect information about software and software dependencies. You can also run agent-based and agentless discovery simultaneously. Use agentless discovery to quickly complete the initial infrastructure assessment and then install agents on select hosts.

Application Discovery Service integrates with application discovery solutions from AWS Partner Network (APN) partners. Third-party application discovery tools can query Application Discovery Service and write to the Application Discovery Service database using a public API. You can then import the data into either a visualization tool or cloud-migration solution.

> **Important**
> Application Discovery Service doesn't gather sensitive information. All data is handled according to the AWS Privacy Policy. You can operate Application Discovery Service offline to inspect collected data before it is shared with the service.

Your AWS account must be granted access to Application Discovery Service, a process called *whitelisting*. This is true for AWS partners and customers alike. To request access, sign up for Application Discovery Service.

This API reference provides descriptions, syntax, and usage examples for each of the actions and data types for Application Discovery Service. The topic for each action shows the API request parameters and the response. Alternatively, you can use one of the AWS SDKs to access an API that is tailored to the programming language or platform that you're using. For more information, see AWS SDKs.

This guide is intended for use with the Application Discovery Service User Guide.

Querying Discovered Configuration Items

A *configuration item* is an IT asset that was discovered in your data center by an agent or the connector. When you use Application Discovery Service, you can specify filters and query specific configuration items for server, application, process, and connection assets.

The tables in the following sections list the available input filters and output sorting options for two Application Discovery Service actions:

- `DescribeConfigurations`
- `ListConfigurations`

The filtering and sorting options are organized by the type of asset to which apply (server, application, process, or connection).

Using the `DescribeConfigurations`Action

The `DescribeConfigurations`action retrieves attributes for a list of configuration IDs. All the supplied IDs must be for the same asset type (server, application, process, or connection). Output fields are specific to the asset type selected. For example, the output for a server configuration item includes a list of attributes about the server, such as host name, operating system, and number of network cards. For more information about command syntax, see DescribeConfigurations.

The `DescribeConfigurations`action does not support filtering.

Output fields for `DescribeConfigurations`

The following tables, organized by asset type, list the supported output fields of the `DescribeConfigurations`action. The ones marked as mandatory are always present in the output.

Server assets

Field	Mandatory
`server.agentId`	
`server.applications`	
`server.applications.hasMoreValues`	
`server.configurationId`	✓
`server.cpuType`	
`server.hostName`	
`server.hypervisor`	
`server.networkInterfaceInfo`	
`server.networkInterfaceInfo.hasMoreValues`	

Field	Mandatory
server.osName	
server.osVersion	
server.tags	
server.tags.hasMoreValues	
server.timeOfCreation	✓
server.type	
server.performance.avgCpuUsagePct	
server.performance.avgDiskReadIOPS	
server.performance.avgDiskReadsPerSecondInKB	
server.performance.avgDiskWriteIOPS	
server.performance.avgDiskWritesPerSecondInKB	
server.performance.avgFreeRAMInKB	
server.performance.avgNetworkReadsPerSecondInKB	
server.performance.avgNetworkWritesPerSecondInKB	
server.performance.maxCpuUsagePct	
server.performance.maxDiskReadIOPS	
server.performance.maxDiskReadsPerSecondInKB	
server.performance.maxDiskWriteIOPS	
server.performance.maxDiskWritesPerSecondInKB	
server.performance.maxNetworkReadsPerSecondInKB	
server.performance.maxNetworkWritesPerSecondInKB	
server.performance.minFreeRAMInKB	
server.performance.numCores	
server.performance.numCpus	
server.performance.numDisks	
server.performance.numNetworkCards	
server.performance.totalRAMInKB	

Process assets

Field	Mandatory
process.commandLine	

Field	Mandatory
process.configurationId	✓
process.name	
process.path	
process.timeOfCreation	✓

Application assets

Field	Mandatory
application.configurationId	✓
application.description	
application.lastModifiedTime	✓
application.name	✓
application.serverCount	✓
application.timeOfCreation	✓

Using the ListConfigurationsAction

The ListConfigurationsaction retrieves a list of configuration items according to the criteria that you specify in a filter. For more information about command syntax, see ListConfigurations.

Output fields for ListConfigurations

The following tables, organized by asset type, list the supported output fields of the ListConfigurationsaction. The ones marked as mandatory are always present in the output.

Server assets

Field	Mandatory
server.configurationId	✓
server.agentId	
server.hostName	
server.osName	
server.osVersion	
server.timeOfCreation	✓
server.type	

Process assets

Field	Mandatory
process.commandLine	
process.configurationId	✓
process.name	
process.path	
process.timeOfCreation	✓
server.agentId	
server.configurationId	✓

Application assets

Field	Mandatory
application.configurationId	✓
application.description	
application.name	✓
application.serverCount	✓
application.timeOfCreation	✓
application.lastModifiedTime	✓

Connection assets

Field	Mandatory
connection.destinationIp	✓
connection.destinationPort	✓
connection.ipVersion	✓
connection.latestTimestamp	✓
connection.occurrence	✓
connection.sourceIp	✓
connection.transportProtocol	
destinationProcess.configurationId	
destinationProcess.name	
destinationServer.configurationId	
destinationServer.hostName	

Field	Mandatory
sourceProcess.configurationId	
sourceProcess.name	
sourceServer.configurationId	
sourceServer.hostName	

Supported filters for `ListConfigurations`

The following tables, organized by asset type, list the supported filters for the `ListConfigurations`action. Filters and values are in a key/value relationship defined by one of the supported logical conditions. You can sort the output of the indicated filters.

Server assets

Filter	Supported conditions	Supported values	Supported sorting
server.configurationId	• EQUALS • NOT_EQUALS • EQ • NE	• Any valid server configuration ID	None
server.hostName	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	• ASC • DESC
server.osName	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	• ASC • DESC
server.osVersion	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	• ASC • DESC
server.agentId	• EQUALS • NOT_EQUALS • EQ • NE	• String	None
server.connectorId	• EQUALS	• String	None

Filter	Supported conditions	Supported values	Supported sorting
	• NOT_EQUALS • EQ • NE		
`server.type`	• EQUALS • NOT_EQUALS • EQ • NE	String with one of the following values: • EC2 • OTHER • VMWARE_VM • VMWARE_HOST • VMWARE_VM_TEMPLATE	None
`server.vmWareInfo.moref`	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	None
`server.vmWareInfo.vcenterUrl`	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	None
`server.vmWareInfo.hostId`	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	None
`server.networkInterfaceInfo.portGroupId`	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	None
`server.networkInterfaceInfo.portGroupName`	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	None

Filter	Supported conditions	Supported values	Supported sorting
server.networkInterfaceInfo.virtualSwitchName	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	None
server.networkInterfaceInfo.ipAddress	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	None
server.networkInterfaceInfo.macAddress	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	None
server.performance.avgCpuUsagePct	• GE • LE • GT • LT	• Percentage	None
server.performance.totalDiskFreeSizeInKB	• GE • LE • GT • LT	• Double	None
server.performance.avgFreeRAMInKB	• GE • LE • GT • LT	• Double	None
server.tag.value	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	None

Filter	Supported conditions	Supported values	Supported sorting
`server.tag.key`	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	None
`server.application.name`	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	None
`server.application.description`	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	None
`server.application.configurationId`	• EQUALS • NOT_EQUALS • EQ • NE	• Any valid application configuration ID	None
`server.process.configurationId`	• EQUALS • NOT_EQUALS • EQ • NE	• ProcessId	None
`server.process.name`	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	None
`server.process.commandLine`	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	None

Application assets

Filter	Supported conditions	Supported values	Supported sorting
application.configurationId	• EQUALS • NOT_EQUALS • EQ • NE	• ApplicationId	None
application.name	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	• ASC • DESC
application.description	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	• ASC • DESC
application.serverCount	Filtering not supported.	Filtering not supported.	• ASC • DESC
application.timeOfCreation	Filtering not supported.	Filtering not supported.	• ASC • DESC
application.lastModifiedTime	Filtering not supported.	Filtering not supported	• ASC • DESC
server.configurationId	• EQUALS • NOT_EQUALS • EQ • NE	• ServerId	None

Process assets

Filter	Supported conditions	Supported values	Supported sorting
process.configurationId	• EQUALS • NOT_EQUALS • EQ • NE	• ProcessId	
process.name	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	• ASC • DESC

Filter	Supported conditions	Supported values	Supported sorting
process.commandLine	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	• ASC • DESC
server.configurationId	• EQUALS • NOT_EQUALS • EQ • NE	• ServerId	
server.hostName	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	• ASC • DESC
server.osName	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	• ASC • DESC
server.osVersion	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	• ASC • DESC
server.agentId	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	

Connection assets

Filter	Supported conditions	Supported values	Supported sorting
connection.sourceIp	• EQUALS • NOT_EQUALS • EQ	• IP	• ASC • DESC

Filter	Supported conditions	Supported values	Supported sorting
	• NE • CONTAINS • NOT_CONTAINS		
`connection.destinationIp`	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• IP	• ASC • DESC
`connection.destinationPort`	• EQUALS • NOT_EQUALS • EQ • NE	• Integer	• ASC • DESC
`sourceServer.configuredId`	• EQUALS • NOT_EQUALS • EQ • NE	• ServerId	
`sourceServer.hostName`	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	• ASC • DESC
`destinationServer.osName`	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	• ASC • DESC
`destinationServer.osVersion`	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	• ASC • DESC
`destinationServer.agentId`	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	

Filter	Supported conditions	Supported values	Supported sorting
`sourceProcess.configurationId`	• EQUALS • NOT_EQUALS • EQ • NE	• ProcessId	
`sourceProcess.name`	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	• ASC • DESC
`sourceProcess.commandLine`	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	• ASC • DESC
`destinationProcess.configurationId`	• EQUALS • NOT_EQUALS • EQ • NE	• ProcessId	
`destinationProcess.name`	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	• ASC • DESC
`destinationprocess.commandLine`	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• String	• ASC • DESC

Actions

The following actions are supported:

AssociateConfigurationItemsToApplication

Associates one or more configuration items with an application.

Request Syntax

```
{
   "applicationConfigurationId": "string",
   "configurationIds": [ "string" ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 80).

The request accepts the following data in JSON format.

applicationConfigurationId (p. 15)

The configuration ID of an application with which items are to be associated.

Type: String

Required: Yes

configurationIds (p. 15)

The ID of each configuration item to be associated with an application.

Type: Array of strings

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 82).

AuthorizationErrorException

The AWS user account does not have permission to perform the action. Check the IAM policy associated with this account.

HTTP Status Code: 400

InvalidParameterException

One or more parameters are not valid. Verify the parameters and try again.

HTTP Status Code: 400

InvalidParameterValueException

The value of one or more parameters are either invalid or out of range. Verify the parameter values and try again.

HTTP Status Code: 400

ServerInternalErrorException

The server experienced an internal error. Try again.

HTTP Status Code: 500

See Also

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

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

CreateApplication

Creates an application with the given name and description.

Request Syntax

```
{
   "description": "string",
   "name": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 80).

The request accepts the following data in JSON format.

description (p. 17)

Description of the application to be created.

Type: String

Required: No

name (p. 17)

Name of the application to be created.

Type: String

Required: Yes

Response Syntax

```
{
   "configurationId": "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.

configurationId (p. 17)

Configuration ID of an application to be created.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 82).

AuthorizationErrorException

The AWS user account does not have permission to perform the action. Check the IAM policy associated with this account.

HTTP Status Code: 400

InvalidParameterException

One or more parameters are not valid. Verify the parameters and try again.

HTTP Status Code: 400

InvalidParameterValueException

The value of one or more parameters are either invalid or out of range. Verify the parameter values and try again.

HTTP Status Code: 400

ServerInternalErrorException

The server experienced an internal error. Try again.

HTTP Status Code: 500

See Also

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

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

CreateTags

Creates one or more tags for configuration items. Tags are metadata that help you categorize IT assets. This API accepts a list of multiple configuration items.

Request Syntax

```
{
   "configurationIds": [ "string" ],
   "tags": [
      {
         "key": "string",
         "value": "string"
      }
   ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 80).

The request accepts the following data in JSON format.

configurationIds (p. 19)

A list of configuration items that you want to tag.

Type: Array of strings

Required: Yes

tags (p. 19)

Tags that you want to associate with one or more configuration items. Specify the tags that you want to create in a *key-value* format. For example:

`{"key": "serverType", "value": "webServer"}`

Type: Array of Tag (p. 78) objects

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 82).

AuthorizationErrorException

The AWS user account does not have permission to perform the action. Check the IAM policy associated with this account.

HTTP Status Code: 400

InvalidParameterException

One or more parameters are not valid. Verify the parameters and try again.

HTTP Status Code: 400

InvalidParameterValueException

The value of one or more parameters are either invalid or out of range. Verify the parameter values and try again.

HTTP Status Code: 400

ResourceNotFoundException

The specified configuration ID was not located. Verify the configuration ID and try again.

HTTP Status Code: 400

ServerInternalErrorException

The server experienced an internal error. Try again.

HTTP Status Code: 500

See Also

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

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

DeleteApplications

Deletes a list of applications and their associations with configuration items.

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 80).

The request accepts the following data in JSON format.

configurationIds (p. 21)

Configuration ID of an application to be deleted.

Type: Array of strings

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 82).

AuthorizationErrorException

The AWS user account does not have permission to perform the action. Check the IAM policy associated with this account.

HTTP Status Code: 400

InvalidParameterException

One or more parameters are not valid. Verify the parameters and try again.

HTTP Status Code: 400

InvalidParameterValueException

The value of one or more parameters are either invalid or out of range. Verify the parameter values and try again.

HTTP Status Code: 400

ServerInternalErrorException

The server experienced an internal error. Try again.

HTTP Status Code: 500

See Also

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

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

DeleteTags

Deletes the association between configuration items and one or more tags. This API accepts a list of multiple configuration items.

Request Syntax

```
{
   "configurationIds": [ "string" ],
   "tags": [
      {
         "key": "string",
         "value": "string"
      }
   ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 80).

The request accepts the following data in JSON format.

configurationIds (p. 23)

A list of configuration items with tags that you want to delete.

Type: Array of strings

Required: Yes

tags (p. 23)

Tags that you want to delete from one or more configuration items. Specify the tags that you want to delete in a *key-value* format. For example:

`{"key": "serverType", "value": "webServer"}`

Type: Array of Tag (p. 78) objects

Required: No

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 82).

AuthorizationErrorException

The AWS user account does not have permission to perform the action. Check the IAM policy associated with this account.

HTTP Status Code: 400

InvalidParameterException

One or more parameters are not valid. Verify the parameters and try again.

HTTP Status Code: 400

InvalidParameterValueException

The value of one or more parameters are either invalid or out of range. Verify the parameter values and try again.

HTTP Status Code: 400

ResourceNotFoundException

The specified configuration ID was not located. Verify the configuration ID and try again.

HTTP Status Code: 400

ServerInternalErrorException

The server experienced an internal error. Try again.

HTTP Status Code: 500

See Also

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

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

DescribeAgents

Lists agents or the Connector by ID or lists all agents/Connectors associated with your user account if you did not specify an ID.

Request Syntax

```
{
   "agentIds": [ "string" ],
   "filters": [
      {
         "condition": "string",
         "name": "string",
         "values": [ "string" ]
      }
   ],
   "maxResults": number,
   "nextToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 80).

The request accepts the following data in JSON format.

agentIds (p. 25)

The agent or the Connector IDs for which you want information. If you specify no IDs, the system returns information about all agents/Connectors associated with your AWS user account.

Type: Array of strings

Required: No

filters (p. 25)

You can filter the request using various logical operators and a *key-value* format. For example:

`{"key": "collectionStatus", "value": "STARTED"}`

Type: Array of Filter (p. 75) objects

Required: No

maxResults (p. 25)

The total number of agents/Connectors to return in a single page of output. The maximum value is 100.

Type: Integer

Required: No

nextToken (p. 25)

Token to retrieve the next set of results. For example, if you previously specified 100 IDs for `DescribeAgentsRequest$agentIds` but set `DescribeAgentsRequest$maxResults` to 10, you received a set of 10 results along with a token. Use that token in this query to get the next set of 10.

Type: String

Required: No

Response Syntax

```
{
   "agentsInfo": [
      {
         "agentId": "string",
         "agentNetworkInfoList": [
            {
               "ipAddress": "string",
               "macAddress": "string"
            }
         ],
         "agentType": "string",
         "collectionStatus": "string",
         "connectorId": "string",
         "health": "string",
         "hostName": "string",
         "lastHealthPingTime": "string",
         "registeredTime": "string",
         "version": "string"
      }
   ],
   "nextToken": "string"
}
```

Response Elements

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

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

agentsInfo (p. 26)

Lists agents or the Connector by ID or lists all agents/Connectors associated with your user account if you did not specify an agent/Connector ID. The output includes agent/Connector IDs, IP addresses, media access control (MAC) addresses, agent/Connector health, host name where the agent/Connector resides, and the version number of each agent/Connector.

Type: Array of AgentInfo (p. 64) objects

nextToken (p. 26)

Token to retrieve the next set of results. For example, if you specified 100 IDs for `DescribeAgentsRequest$agentIds` but set `DescribeAgentsRequest$maxResults` to 10, you received a set of 10 results along with this token. Use this token in the next query to retrieve the next set of 10.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 82).

AuthorizationErrorException

The AWS user account does not have permission to perform the action. Check the IAM policy associated with this account.

HTTP Status Code: 400

InvalidParameterException

One or more parameters are not valid. Verify the parameters and try again.

HTTP Status Code: 400

InvalidParameterValueException

The value of one or more parameters are either invalid or out of range. Verify the parameter values and try again.

HTTP Status Code: 400

ServerInternalErrorException

The server experienced an internal error. Try again.

HTTP Status Code: 500

See Also

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

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

DescribeConfigurations

Retrieves attributes for a list of configuration item IDs. All of the supplied IDs must be for the same asset type (server, application, process, or connection). Output fields are specific to the asset type selected. For example, the output for a *server* configuration item includes a list of attributes about the server, such as host name, operating system, and number of network cards.

For a complete list of outputs for each asset type, see Using the DescribeConfigurations Action.

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 80).

The request accepts the following data in JSON format.

configurationIds (p. 28)

One or more configuration IDs.

Type: Array of strings

Required: Yes

Response Syntax

```
{
   "configurations": [
      {
         "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.

configurations (p. 28)

A key in the response map. The value is an array of data.

Type: Array of string to string maps

Errors

For information about the errors that are common to all actions, see Common Errors (p. 82).

AuthorizationErrorException

The AWS user account does not have permission to perform the action. Check the IAM policy associated with this account.

HTTP Status Code: 400

InvalidParameterException

One or more parameters are not valid. Verify the parameters and try again.

HTTP Status Code: 400

InvalidParameterValueException

The value of one or more parameters are either invalid or out of range. Verify the parameter values and try again.

HTTP Status Code: 400

ServerInternalErrorException

The server experienced an internal error. Try again.

HTTP Status Code: 500

See Also

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

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

DescribeExportConfigurations

Deprecated. Use `DescribeExportTasks` instead.

Retrieves the status of a given export process. You can retrieve status from a maximum of 100 processes.

Request Syntax

```
{
   "exportIds": [ "string" ],
   "maxResults": number,
   "nextToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 80).

The request accepts the following data in JSON format.

exportIds (p. 30)

A unique identifier that you can use to query the export status.

Type: Array of strings

Required: No

maxResults (p. 30)

The maximum number of results that you want to display as a part of the query.

Type: Integer

Required: No

nextToken (p. 30)

A token to get the next set of results. For example, if you specify 100 IDs for `DescribeExportConfigurationsRequest$exportIds` but set `DescribeExportConfigurationsRequest$maxResults` to 10, you get results in a set of 10. Use the token in the query to get the next set of 10.

Type: String

Required: No

Response Syntax

```
{
   "exportsInfo": [
      {
         "configurationsDownloadUrl": "string",
         "exportId": "string",
         "exportRequestTime": number,
```

```
        "exportStatus": "string",
        "isTruncated": boolean,
        "requestedEndTime": number,
        "requestedStartTime": number,
        "statusMessage": "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.

exportsInfo (p. 30)

Returns export details. When the status is complete, the response includes a URL for an Amazon S3 bucket where you can view the data in a CSV file.

Type: Array of ExportInfo (p. 73) objects

nextToken (p. 30)

A token to get the next set of results. For example, if you specify 100 IDs for `DescribeExportConfigurationsRequest$exportIds` but set `DescribeExportConfigurationsRequest$maxResults` to 10, you get results in a set of 10. Use the token in the query to get the next set of 10.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 82).

AuthorizationErrorException

The AWS user account does not have permission to perform the action. Check the IAM policy associated with this account.

HTTP Status Code: 400

InvalidParameterException

One or more parameters are not valid. Verify the parameters and try again.

HTTP Status Code: 400

InvalidParameterValueException

The value of one or more parameters are either invalid or out of range. Verify the parameter values and try again.

HTTP Status Code: 400

ResourceNotFoundException

The specified configuration ID was not located. Verify the configuration ID and try again.

HTTP Status Code: 400

ServerInternalErrorException

The server experienced an internal error. Try again.

HTTP Status Code: 500

See Also

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

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

DescribeExportTasks

Retrieve status of one or more export tasks. You can retrieve the status of up to 100 export tasks.

Request Syntax

```
{
   "exportIds": [ "string" ],
   "filters": [
      {
         "condition": "string",
         "name": "string",
         "values": [ "string" ]
      }
   ],
   "maxResults": number,
   "nextToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 80).

The request accepts the following data in JSON format.

exportIds (p. 33)

One or more unique identifiers used to query the status of an export request.

Type: Array of strings

Required: No

filters (p. 33)

One or more filters.

- `AgentId` - ID of the agent whose collected data will be exported

Type: Array of ExportFilter (p. 72) objects

Required: No

maxResults (p. 33)

The maximum number of volume results returned by `DescribeExportTasks` in paginated output. When this parameter is used, `DescribeExportTasks` only returns `maxResults` results in a single page along with a `nextToken` response element.

Type: Integer

Required: No

nextToken (p. 33)

The `nextToken` value returned from a previous paginated `DescribeExportTasks` request where `maxResults` was used and the results exceeded the value of that parameter. Pagination continues from the end of the previous results that returned the `nextToken` value. This value is null when there are no more results to return.

Type: String

Required: No

Response Syntax

```
{
   "exportsInfo": [
      {
         "configurationsDownloadUrl": "string",
         "exportId": "string",
         "exportRequestTime": number,
         "exportStatus": "string",
         "isTruncated": boolean,
         "requestedEndTime": number,
         "requestedStartTime": number,
         "statusMessage": "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.

exportsInfo (p. 34)

Contains one or more sets of export request details. When the status of a request is `SUCCEEDED`, the response includes a URL for an Amazon S3 bucket where you can view the data in a CSV file.

Type: Array of ExportInfo (p. 73) objects

nextToken (p. 34)

The `nextToken` value to include in a future `DescribeExportTasks` request. When the results of a `DescribeExportTasks` request exceed `maxResults`, this value can be used to retrieve the next page of results. This value is null when there are no more results to return.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 82).

AuthorizationErrorException

The AWS user account does not have permission to perform the action. Check the IAM policy associated with this account.

HTTP Status Code: 400

InvalidParameterException

One or more parameters are not valid. Verify the parameters and try again.

HTTP Status Code: 400

InvalidParameterValueException

The value of one or more parameters are either invalid or out of range. Verify the parameter values and try again.

HTTP Status Code: 400

ServerInternalErrorException

The server experienced an internal error. Try again.

HTTP Status Code: 500

Example

In the following example or examples, the Authorization header contents (`AUTHPARAMS`) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see Signature Version 4 Signing Process in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the AWS Command Line Interface (AWS CLI) or one of the AWS SDKs to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Retrieve status of two specified export tasks

Sample Request

```
POST / HTTP/1.1
Host: discovery.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 109
X-Amz-Target: AWSPoseidonService_V2015_11_01.DescribeExportTasks
X-Amz-Date: 20170308T232123Z
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{
   "exportIds":[
      "export-123a4b56-7c89-01d2-3ef4-example5678f",
      "export-654a3b21-7c89-01d2-3ef4-example8765f"
   ]
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 123a4b56-7c89-01d2-3ef4-example5678f
Content-Type: application/x-amz-json-1.1
Content-Length: 1140
Date: Wed, 08 Mar 2017 23:21:25 GMT

{
   "exportsInfo":[
      {
         "configurationsDownloadUrl":"[URL]",
         "exportId":"export-123a4b56-7c89-01d2-3ef4-example5678f",
         "exportRequestTime":1.489001254713E9,
         "exportStatus":"SUCCEEDED",
```

```
         "statusMessage":"Data export ran successfully and is accessible from the download
 URL. The URL will expire in 24 hours. The export data expires in 10 days."
      },
      {
         "configurationsDownloadUrl":"[URL]",
         "exportId":"export-654a3b21-7c89-01d2-3ef4-example8765f",
         "exportRequestTime":1.488920016713E9,
         "exportStatus":"SUCCEEDED",
         "statusMessage":"Data export ran successfully and is accessible from the download
 URL. The URL will expire in 24 hours. The export data expires in 10 days."
      }
   ],
   "nextToken":""
}
```


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
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

DescribeTags

Retrieves a list of configuration items that are tagged with a specific tag. Or retrieves a list of all tags assigned to a specific configuration item.

Request Syntax

```
{
   "filters": [
      {
         "name": "string",
         "values": [ "string" ]
      }
   ],
   "maxResults": number,
   "nextToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 80).

The request accepts the following data in JSON format.

filters (p. 37)

You can filter the list using a *key-value* format. You can separate these items by using logical operators. Allowed filters include `tagKey`, `tagValue`, and `configurationId`.

Type: Array of TagFilter (p. 79) objects

Required: No

maxResults (p. 37)

The total number of items to return in a single page of output. The maximum value is 100.

Type: Integer

Required: No

nextToken (p. 37)

A token to start the list. Use this token to get the next set of results.

Type: String

Required: No

Response Syntax

```
{
   "nextToken": "string",
   "tags": [
      {
         "configurationId": "string",
```

```
        "configurationType": "string",
        "key": "string",
        "timeOfCreation": number,
        "value": "string"
      }
   ]
}
```

Response Elements

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

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

nextToken (p. 37)

The call returns a token. Use this token to get the next set of results.

Type: String

tags (p. 37)

Depending on the input, this is a list of configuration items tagged with a specific tag, or a list of tags for a specific configuration item.

Type: Array of ConfigurationTag (p. 67) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 82).

AuthorizationErrorException

The AWS user account does not have permission to perform the action. Check the IAM policy associated with this account.

HTTP Status Code: 400

InvalidParameterException

One or more parameters are not valid. Verify the parameters and try again.

HTTP Status Code: 400

InvalidParameterValueException

The value of one or more parameters are either invalid or out of range. Verify the parameter values and try again.

HTTP Status Code: 400

ResourceNotFoundException

The specified configuration ID was not located. Verify the configuration ID and try again.

HTTP Status Code: 400

ServerInternalErrorException

The server experienced an internal error. Try again.

HTTP Status Code: 500

See Also

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

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

DisassociateConfigurationItemsFromApplication

Disassociates one or more configuration items from an application.

Request Syntax

```
{
   "applicationConfigurationId": "string",
   "configurationIds": [ "string" ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 80).

The request accepts the following data in JSON format.

applicationConfigurationId (p. 40)

Configuration ID of an application from which each item is disassociated.

Type: String

Required: Yes

configurationIds (p. 40)

Configuration ID of each item to be disassociated from an application.

Type: Array of strings

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 82).

AuthorizationErrorException

The AWS user account does not have permission to perform the action. Check the IAM policy associated with this account.

HTTP Status Code: 400

InvalidParameterException

One or more parameters are not valid. Verify the parameters and try again.

HTTP Status Code: 400

InvalidParameterValueException

The value of one or more parameters are either invalid or out of range. Verify the parameter values and try again.

HTTP Status Code: 400

ServerInternalErrorException

The server experienced an internal error. Try again.

HTTP Status Code: 500

See Also

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

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

ExportConfigurations

Deprecated. Use `StartExportTask` instead.

Exports all discovered configuration data to an Amazon S3 bucket or an application that enables you to view and evaluate the data. Data includes tags and tag associations, processes, connections, servers, and system performance. This API returns an export ID that you can query using the *DescribeExportConfigurations* API. The system imposes a limit of two configuration exports in six hours.

Response Syntax

```
{
   "exportId": "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.

exportId (p. 42)

A unique identifier that you can use to query the export status.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 82).

AuthorizationErrorException

The AWS user account does not have permission to perform the action. Check the IAM policy associated with this account.

HTTP Status Code: 400

InvalidParameterException

One or more parameters are not valid. Verify the parameters and try again.

HTTP Status Code: 400

InvalidParameterValueException

The value of one or more parameters are either invalid or out of range. Verify the parameter values and try again.

HTTP Status Code: 400

OperationNotPermittedException

This operation is not permitted.

HTTP Status Code: 400

ServerInternalErrorException

The server experienced an internal error. Try again.

HTTP Status Code: 500

See Also

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

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

GetDiscoverySummary

Retrieves a short summary of discovered assets.

Response Syntax

```
{
   "agentSummary": {
      "activeAgents": number,
      "blackListedAgents": number,
      "healthyAgents": number,
      "shutdownAgents": number,
      "totalAgents": number,
      "unhealthyAgents": number,
      "unknownAgents": number
   },
   "applications": number,
   "connectorSummary": {
      "activeConnectors": number,
      "blackListedConnectors": number,
      "healthyConnectors": number,
      "shutdownConnectors": number,
      "totalConnectors": number,
      "unhealthyConnectors": number,
      "unknownConnectors": number
   },
   "servers": number,
   "serversMappedToApplications": number,
   "serversMappedtoTags": number
}
```

Response Elements

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

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

agentSummary (p. 44)

Details about discovered agents, including agent status and health.

Type: CustomerAgentInfo (p. 68) object

applications (p. 44)

The number of applications discovered.

Type: Long

connectorSummary (p. 44)

Details about discovered connectors, including connector status and health.

Type: CustomerConnectorInfo (p. 70) object

servers (p. 44)

The number of servers discovered.

Type: Long

serversMappedToApplications (p. 44)

The number of servers mapped to applications.

Type: Long

serversMappedtoTags (p. 44)

The number of servers mapped to tags.

Type: Long

Errors

For information about the errors that are common to all actions, see Common Errors (p. 82).

AuthorizationErrorException

The AWS user account does not have permission to perform the action. Check the IAM policy associated with this account.

HTTP Status Code: 400

InvalidParameterException

One or more parameters are not valid. Verify the parameters and try again.

HTTP Status Code: 400

InvalidParameterValueException

The value of one or more parameters are either invalid or out of range. Verify the parameter values and try again.

HTTP Status Code: 400

ServerInternalErrorException

The server experienced an internal error. Try again.

HTTP Status Code: 500

See Also

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

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

ListConfigurations

Retrieves a list of configuration items according to criteria that you specify in a filter. The filter criteria identifies the relationship requirements.

Request Syntax

```
{
   "configurationType": "string",
   "filters": [
      {
         "condition": "string",
         "name": "string",
         "values": [ "string" ]
      }
   ],
   "maxResults": number,
   "nextToken": "string",
   "orderBy": [
      {
         "fieldName": "string",
         "sortOrder": "string"
      }
   ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 80).

The request accepts the following data in JSON format.

configurationType (p. 46)

A valid configuration identified by Application Discovery Service.

Type: String

Valid Values: `SERVER | PROCESS | CONNECTION | APPLICATION`

Required: Yes

filters (p. 46)

You can filter the request using various logical operators and a *key-value* format. For example:

`{"key": "serverType", "value": "webServer"}`

For a complete list of filter options and guidance about using them with this action, see Querying Discovered Configuration Items.

Type: Array of Filter (p. 75) objects

Required: No

maxResults (p. 46)

The total number of items to return. The maximum value is 100.

Type: Integer

Required: No

nextToken (p. 46)

Token to retrieve the next set of results. For example, if a previous call to ListConfigurations returned 100 items, but you set `ListConfigurationsRequest$maxResults` to 10, you received a set of 10 results along with a token. Use that token in this query to get the next set of 10.

Type: String

Required: No

orderBy (p. 46)

Certain filter criteria return output that can be sorted in ascending or descending order. For a list of output characteristics for each filter, see Using the ListConfigurations Action.

Type: Array of OrderByElement (p. 77) objects

Required: No

Response Syntax

```
{
   "configurations": [
      {
         "string" : "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.

configurations (p. 47)

Returns configuration details, including the configuration ID, attribute names, and attribute values.

Type: Array of string to string maps

nextToken (p. 47)

Token to retrieve the next set of results. For example, if your call to ListConfigurations returned 100 items, but you set `ListConfigurationsRequest$maxResults` to 10, you received a set of 10 results along with this token. Use this token in the next query to retrieve the next set of 10.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 82).

AuthorizationErrorException

The AWS user account does not have permission to perform the action. Check the IAM policy associated with this account.

HTTP Status Code: 400

InvalidParameterException

One or more parameters are not valid. Verify the parameters and try again.

HTTP Status Code: 400

InvalidParameterValueException

The value of one or more parameters are either invalid or out of range. Verify the parameter values and try again.

HTTP Status Code: 400

ResourceNotFoundException

The specified configuration ID was not located. Verify the configuration ID and try again.

HTTP Status Code: 400

ServerInternalErrorException

The server experienced an internal error. Try again.

HTTP Status Code: 500

See Also

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

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

ListServerNeighbors

Retrieves a list of servers that are one network hop away from a specified server.

Request Syntax

```
{
   "configurationId": "string",
   "maxResults": number,
   "neighborConfigurationIds": [ "string" ],
   "nextToken": "string",
   "portInformationNeeded": boolean
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 80).

The request accepts the following data in JSON format.

configurationId (p. 49)

Configuration ID of the server for which neighbors are being listed.

Type: String

Required: Yes

maxResults (p. 49)

Maximum number of results to return in a single page of output.

Type: Integer

Required: No

neighborConfigurationIds (p. 49)

List of configuration IDs to test for one-hop-away.

Type: Array of strings

Required: No

nextToken (p. 49)

Token to retrieve the next set of results. For example, if you previously specified 100 IDs for `ListServerNeighborsRequest$neighborConfigurationIds` but set `ListServerNeighborsRequest$maxResults` to 10, you received a set of 10 results along with a token. Use that token in this query to get the next set of 10.

Type: String

Required: No

portInformationNeeded (p. 49)

Flag to indicate if port and protocol information is needed as part of the response.

Type: Boolean

Required: No

Response Syntax

```
{
   "knownDependencyCount": number,
   "neighbors": [
      {
         "connectionsCount": number,
         "destinationPort": number,
         "destinationServerId": "string",
         "sourceServerId": "string",
         "transportProtocol": "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.

knownDependencyCount (p. 50)

Count of distinct servers that are one hop away from the given server.

Type: Long

neighbors (p. 50)

List of distinct servers that are one hop away from the given server.

Type: Array of NeighborConnectionDetail (p. 76) objects

nextToken (p. 50)

Token to retrieve the next set of results. For example, if you specified 100 IDs for `ListServerNeighborsRequest$neighborConfigurationIds` but set `ListServerNeighborsRequest$maxResults` to 10, you received a set of 10 results along with this token. Use this token in the next query to retrieve the next set of 10.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 82).

AuthorizationErrorException

The AWS user account does not have permission to perform the action. Check the IAM policy associated with this account.

HTTP Status Code: 400

InvalidParameterException

One or more parameters are not valid. Verify the parameters and try again.

HTTP Status Code: 400

InvalidParameterValueException

The value of one or more parameters are either invalid or out of range. Verify the parameter values and try again.

HTTP Status Code: 400

ServerInternalErrorException

The server experienced an internal error. Try again.

HTTP Status Code: 500

See Also

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

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

StartDataCollectionByAgentIds

Instructs the specified agents or connectors to start collecting data.

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 80).

The request accepts the following data in JSON format.

agentIds (p. 52)

The IDs of the agents or connectors from which to start collecting data. If you send a request to an agent/connector ID that you do not have permission to contact, according to your AWS account, the service does not throw an exception. Instead, it returns the error in the *Description* field. If you send a request to multiple agents/connectors and you do not have permission to contact some of those agents/connectors, the system does not throw an exception. Instead, the system shows `Failed` in the *Description* field.

Type: Array of strings

Required: Yes

Response Syntax

```
{
   "agentsConfigurationStatus": [
      {
         "agentId": "string",
         "description": "string",
         "operationSucceeded": boolean
      }
   ]
}
```

Response Elements

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

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

agentsConfigurationStatus (p. 52)

Information about agents or the connector that were instructed to start collecting data. Information includes the agent/connector ID, a description of the operation performed, and whether the agent/connector configuration was updated.

Type: Array of AgentConfigurationStatus (p. 63) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 82).

AuthorizationErrorException

The AWS user account does not have permission to perform the action. Check the IAM policy associated with this account.

HTTP Status Code: 400

InvalidParameterException

One or more parameters are not valid. Verify the parameters and try again.

HTTP Status Code: 400

InvalidParameterValueException

The value of one or more parameters are either invalid or out of range. Verify the parameter values and try again.

HTTP Status Code: 400

ServerInternalErrorException

The server experienced an internal error. Try again.

HTTP Status Code: 500

See Also

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

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

StartExportTask

Begins the export of discovered data to an S3 bucket.

If you specify `agentIds` in a filter, the task exports up to 72 hours of detailed data collected by the identified Application Discovery Agent, including network, process, and performance details. A time range for exported agent data may be set by using `startTime` and `endTime`. Export of detailed agent data is limited to five concurrently running exports.

If you do not include an `agentIds` filter, summary data is exported that includes both AWS Agentless Discovery Connector data and summary data from AWS Discovery Agents. Export of summary data is limited to two exports per day.

Request Syntax

```
{
   "endTime": number,
   "exportDataFormat": [ "string" ],
   "filters": [
      {
         "condition": "string",
         "name": "string",
         "values": [ "string" ]
      }
   ],
   "startTime": number
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 80).

The request accepts the following data in JSON format.

endTime (p. 54)

The end timestamp for exported data from the single Application Discovery Agent selected in the filters. If no value is specified, exported data includes the most recent data collected by the agent.

Type: Timestamp

Required: No

exportDataFormat (p. 54)

The file format for the returned export data. Default value is `CSV`. **Note:** *The* `GRAPHML` *option has been deprecated.*

Type: Array of strings

Valid Values: `CSV | GRAPHML`

Required: No

filters (p. 54)

If a filter is present, it selects the single `agentId` of the Application Discovery Agent for which data is exported. The `agentId` can be found in the results of the `DescribeAgents` API or CLI. If no

filter is present, `startTime` and `endTime` are ignored and exported data includes both Agentless Discovery Connector data and summary data from Application Discovery agents.

Type: Array of ExportFilter (p. 72) objects

Required: No

startTime (p. 54)

The start timestamp for exported data from the single Application Discovery Agent selected in the filters. If no value is specified, data is exported starting from the first data collected by the agent.

Type: Timestamp

Required: No

Response Syntax

```
{
   "exportId": "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.

exportId (p. 55)

A unique identifier used to query the status of an export request.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 82).

AuthorizationErrorException

The AWS user account does not have permission to perform the action. Check the IAM policy associated with this account.

HTTP Status Code: 400

InvalidParameterException

One or more parameters are not valid. Verify the parameters and try again.

HTTP Status Code: 400

InvalidParameterValueException

The value of one or more parameters are either invalid or out of range. Verify the parameter values and try again.

HTTP Status Code: 400

OperationNotPermittedException

This operation is not permitted.

HTTP Status Code: 400

ServerInternalErrorException

The server experienced an internal error. Try again.

HTTP Status Code: 500

Example

In the following example or examples, the Authorization header contents (`AUTHPARAMS`) must be replaced with an AWS Signature Version 4 signature. For more information about creating these signatures, see Signature Version 4 Signing Process in the *AWS General Reference*.

You only need to learn how to sign HTTP requests if you intend to manually create them. When you use the AWS Command Line Interface (AWS CLI) or one of the AWS SDKs to make requests to AWS, these tools automatically sign the requests for you with the access key that you specify when you configure the tools. When you use these tools, you don't need to learn how to sign requests yourself.

Start an export task of summary data filtering for an agentId and a time range

Sample Request

```
POST / HTTP/1.1
Host: discovery.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 141
X-Amz-Target: AWSPoseidonService_V2015_11_01.StartExportTask
X-Amz-Date: 20170711T004454Z
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{
   "endTime":1499601600,
   "startTime":1499428800,
   "filters":[
      {
         "values":[
            "i-06ea7f227a9ad1b86"
         ],
         "name":"agentIds",
         "condition":"EQUALS"
      }
   ]
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 123a4b56-7c89-01d2-3ef4-example5678f
Content-Type: application/x-amz-json-1.1
Content-Length: 58
Date: Tue, 11 Jul 2017 00:44:54 GMT
```

```
{
   "exportId":"export-123a4b56-7c89-01d2-3ef4-example5678f"
}
```

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
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

StopDataCollectionByAgentIds

Instructs the specified agents or connectors to stop collecting data.

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 80).

The request accepts the following data in JSON format.

agentIds (p. 58)

The IDs of the agents or connectors from which to stop collecting data.

Type: Array of strings

Required: Yes

Response Syntax

```
{
   "agentsConfigurationStatus": [
      {
         "agentId": "string",
         "description": "string",
         "operationSucceeded": boolean
      }
   ]
}
```

Response Elements

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

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

agentsConfigurationStatus (p. 58)

Information about the agents or connector that were instructed to stop collecting data. Information includes the agent/connector ID, a description of the operation performed, and whether the agent/connector configuration was updated.

Type: Array of AgentConfigurationStatus (p. 63) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 82).

AuthorizationErrorException

The AWS user account does not have permission to perform the action. Check the IAM policy associated with this account.

HTTP Status Code: 400

InvalidParameterException

One or more parameters are not valid. Verify the parameters and try again.

HTTP Status Code: 400

InvalidParameterValueException

The value of one or more parameters are either invalid or out of range. Verify the parameter values and try again.

HTTP Status Code: 400

ServerInternalErrorException

The server experienced an internal error. Try again.

HTTP Status Code: 500

See Also

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

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

UpdateApplication

Updates metadata about an application.

Request Syntax

```
{
   "configurationId": "string",
   "description": "string",
   "name": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 80).

The request accepts the following data in JSON format.

configurationId (p. 60)

Configuration ID of the application to be updated.

Type: String

Required: Yes

description (p. 60)

New description of the application to be updated.

Type: String

Required: No

name (p. 60)

New name of the application to be updated.

Type: String

Required: No

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 82).

AuthorizationErrorException

The AWS user account does not have permission to perform the action. Check the IAM policy associated with this account.

HTTP Status Code: 400

InvalidParameterException

One or more parameters are not valid. Verify the parameters and try again.

HTTP Status Code: 400

InvalidParameterValueException

The value of one or more parameters are either invalid or out of range. Verify the parameter values and try again.

HTTP Status Code: 400

ServerInternalErrorException

The server experienced an internal error. Try again.

HTTP Status Code: 500

See Also

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

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

Data Types

The AWS Application Discovery Service 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:

AgentConfigurationStatus

Information about agents or connectors that were instructed to start collecting data. Information includes the agent/connector ID, a description of the operation, and whether the agent/connector configuration was updated.

Contents

agentId

The agent/connector ID.

Type: String

Required: No

description

A description of the operation performed.

Type: String

Required: No

operationSucceeded

Information about the status of the `StartDataCollection` and `StopDataCollection` operations. The system has recorded the data collection operation. The agent/connector receives this command the next time it polls for a new command.

Type: Boolean

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
- AWS SDK for Ruby V2

AgentInfo

Information about agents or connectors associated with the user's AWS account. Information includes agent or connector IDs, IP addresses, media access control (MAC) addresses, agent or connector health, hostname where the agent or connector resides, and agent version for each agent.

Contents

agentId

The agent or connector ID.

Type: String

Required: No

agentNetworkInfoList

Network details about the host where the agent or connector resides.

Type: Array of AgentNetworkInfo (p. 66) objects

Required: No

agentType

Type of agent.

Type: String

Required: No

collectionStatus

Status of the collection process for an agent or connector.

Type: String

Required: No

connectorId

The ID of the connector.

Type: String

Required: No

health

The health of the agent or connector.

Type: String

Valid Values: `HEALTHY | UNHEALTHY | RUNNING | UNKNOWN | BLACKLISTED | SHUTDOWN`

Required: No

hostName

The name of the host where the agent or connector resides. The host can be a server or virtual machine.

Type: String

Required: No

lastHealthPingTime

Time since agent or connector health was reported.

Type: String

Required: No

registeredTime

Agent's first registration timestamp in UTC.

Type: String

Required: No

version

The agent or connector version.

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
- AWS SDK for Ruby V2

AgentNetworkInfo

Network details about the host where the agent/connector resides.

Contents

ipAddress

The IP address for the host where the agent/connector resides.

Type: String

Required: No

macAddress

The MAC address for the host where the agent/connector resides.

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
- AWS SDK for Ruby V2

ConfigurationTag

Tags for a configuration item. Tags are metadata that help you categorize IT assets.

Contents

configurationId

The configuration ID for the item to tag. You can specify a list of keys and values.

Type: String

Required: No

configurationType

A type of IT asset to tag.

Type: String

Valid Values: `SERVER | PROCESS | CONNECTION | APPLICATION`

Required: No

key

A type of tag on which to filter. For example, *serverType*.

Type: String

Required: No

timeOfCreation

The time the configuration tag was created in Coordinated Universal Time (UTC).

Type: Timestamp

Required: No

value

A value on which to filter. For example *key = serverType* and *value = web server*.

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
- AWS SDK for Ruby V2

CustomerAgentInfo

Inventory data for installed discovery agents.

Contents

activeAgents

Number of active discovery agents.

Type: Integer

Required: Yes

blackListedAgents

Number of blacklisted discovery agents.

Type: Integer

Required: Yes

healthyAgents

Number of healthy discovery agents

Type: Integer

Required: Yes

shutdownAgents

Number of discovery agents with status SHUTDOWN.

Type: Integer

Required: Yes

totalAgents

Total number of discovery agents.

Type: Integer

Required: Yes

unhealthyAgents

Number of unhealthy discovery agents.

Type: Integer

Required: Yes

unknownAgents

Number of unknown discovery agents.

Type: Integer

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
- AWS SDK for Ruby V2

CustomerConnectorInfo

Inventory data for installed discovery connectors.

Contents

activeConnectors

Number of active discovery connectors.

Type: Integer

Required: Yes

blackListedConnectors

Number of blacklisted discovery connectors.

Type: Integer

Required: Yes

healthyConnectors

Number of healthy discovery connectors.

Type: Integer

Required: Yes

shutdownConnectors

Number of discovery connectors with status SHUTDOWN,

Type: Integer

Required: Yes

totalConnectors

Total number of discovery connectors.

Type: Integer

Required: Yes

unhealthyConnectors

Number of unhealthy discovery connectors.

Type: Integer

Required: Yes

unknownConnectors

Number of unknown discovery connectors.

Type: Integer

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
- AWS SDK for Ruby V2

ExportFilter

Used to select which agent's data is to be exported. A single agent ID may be selected for export using the StartExportTask action.

Contents

condition

Supported condition: `EQUALS`

Type: String

Required: Yes

name

A single `ExportFilter` name. Supported filters: `agentId`.

Type: String

Required: Yes

values

A single `agentId` for a Discovery Agent. An `agentId` can be found using the DescribeAgents action. Typically an ADS `agentId` is in the form `o-0123456789abcdef0`.

Type: Array of strings

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
- AWS SDK for Ruby V2

ExportInfo

Information regarding the export status of discovered data. The value is an array of objects.

Contents

configurationsDownloadUrl

A URL for an Amazon S3 bucket where you can review the exported data. The URL is displayed only if the export succeeded.

Type: String

Required: No

exportId

A unique identifier used to query an export.

Type: String

Required: Yes

exportRequestTime

The time that the data export was initiated.

Type: Timestamp

Required: Yes

exportStatus

The status of the data export job.

Type: String

Valid Values: `FAILED | SUCCEEDED | IN_PROGRESS`

Required: Yes

isTruncated

If true, the export of agent information exceeded the size limit for a single export and the exported data is incomplete for the requested time range. To address this, select a smaller time range for the export by using `startDate` and `endDate`.

Type: Boolean

Required: No

requestedEndTime

The `endTime` used in the `StartExportTask` request. If no `endTime` was requested, this result does not appear in `ExportInfo`.

Type: Timestamp

Required: No

requestedStartTime

The value of `startTime` parameter in the `StartExportTask` request. If no `startTime` was requested, this result does not appear in `ExportInfo`.

Type: Timestamp

Required: No

statusMessage

A status message provided for API callers.

Type: String

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
- AWS SDK for Ruby V2

Filter

A filter that can use conditional operators.

For more information about filters, see Querying Discovered Configuration Items.

Contents

condition

A conditional operator. The following operators are valid: EQUALS, NOT_EQUALS, CONTAINS, NOT_CONTAINS. If you specify multiple filters, the system utilizes all filters as though concatenated by *AND*. If you specify multiple values for a particular filter, the system differentiates the values using *OR*. Calling either *DescribeConfigurations* or *ListConfigurations* returns attributes of matching configuration items.

Type: String

Required: Yes

name

The name of the filter.

Type: String

Required: Yes

values

A string value on which to filter. For example, if you choose the `destinationServer.osVersion` filter name, you could specify `Ubuntu` for the value.

Type: Array of strings

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
- AWS SDK for Ruby V2

NeighborConnectionDetail

Details about neighboring servers.

Contents

connectionsCount

The number of open network connections with the neighboring server.

Type: Long

Required: Yes

destinationPort

The destination network port for the connection.

Type: Integer

Required: No

destinationServerId

The ID of the server that accepted the network connection.

Type: String

Required: Yes

sourceServerId

The ID of the server that opened the network connection.

Type: String

Required: Yes

transportProtocol

The network protocol used for the connection.

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
- AWS SDK for Ruby V2

OrderByElement

A field and direction for ordered output.

Contents

fieldName

The field on which to order.

Type: String

Required: Yes

sortOrder

Ordering direction.

Type: String

Valid Values: `ASC | DESC`

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
- AWS SDK for Ruby V2

Tag

Metadata that help you categorize IT assets.

Contents

key

The type of tag on which to filter.

Type: String

Required: Yes

value

A value for a tag key on which to filter.

Type: String

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
- AWS SDK for Ruby V2

TagFilter

The tag filter. Valid names are: `tagKey`, `tagValue`, `configurationId`.

Contents

name

A name of the tag filter.

Type: String

Required: Yes

values

Values for the tag filter.

Type: Array of strings

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
- AWS SDK for Ruby V2

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

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