# Table of Contents

Welcome ........................................................................................................................................... 1
Querying Discovered Configuration Items .................................................................................................................. 2
- Using the DescribeConfigurationsAction .................................................................................................................. 2
- Using the ListConfigurationsAction ........................................................................................................................ 4
Actions .............................................................................................................................................. 14
- AssociateConfigurationItemsToApplication ........................................................................................................ 15
  - Request Syntax .................................................................................................................................................. 15
  - Request Parameters ........................................................................................................................................... 15
  - Response Elements ........................................................................................................................................... 15
  - Errors ................................................................................................................................................................. 15
  - Example .............................................................................................................................................................. 16
  - See Also ............................................................................................................................................................. 16
- CreateApplication .................................................................................................................................................. 17
  - Request Syntax .................................................................................................................................................. 17
  - Request Parameters ........................................................................................................................................... 17
  - Response Syntax ............................................................................................................................................... 17
  - Response Elements ........................................................................................................................................... 17
  - Errors ................................................................................................................................................................. 17
  - Example .............................................................................................................................................................. 18
  - See Also ............................................................................................................................................................. 18
- CreateTags ........................................................................................................................................................ 20
  - Request Syntax .................................................................................................................................................. 20
  - Request Parameters ........................................................................................................................................... 20
  - Response Elements ........................................................................................................................................... 20
  - Errors ................................................................................................................................................................. 20
  - Example .............................................................................................................................................................. 21
  - See Also ............................................................................................................................................................. 21
- DeleteApplications ................................................................................................................................................ 23
  - Request Syntax .................................................................................................................................................. 23
  - Request Parameters ........................................................................................................................................... 23
  - Response Elements ........................................................................................................................................... 23
  - Errors ................................................................................................................................................................. 23
  - Example .............................................................................................................................................................. 24
  - See Also ............................................................................................................................................................. 24
- DeleteTags ........................................................................................................................................................ 25
  - Request Syntax .................................................................................................................................................. 25
  - Request Parameters ........................................................................................................................................... 25
  - Response Elements ........................................................................................................................................... 25
  - Errors ................................................................................................................................................................. 25
  - Example .............................................................................................................................................................. 26
  - See Also ............................................................................................................................................................. 26
- DescribeAgents ................................................................................................................................................... 28
  - Request Syntax .................................................................................................................................................. 28
  - Request Parameters ........................................................................................................................................... 28
  - Response Syntax ............................................................................................................................................... 28
  - Response Elements ........................................................................................................................................... 28
  - Errors ................................................................................................................................................................. 29
  - Example .............................................................................................................................................................. 30
  - See Also ............................................................................................................................................................. 30
- DescribeConfigurations .......................................................................................................................................... 32
  - Request Syntax .................................................................................................................................................. 32
  - Request Parameters ........................................................................................................................................... 32
  - Response Syntax ............................................................................................................................................... 32
  - Response Elements ........................................................................................................................................... 32
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 the following 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 DescribeConfigurationsAction

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

<table>
<thead>
<tr>
<th>Field</th>
<th>Mandatory</th>
</tr>
</thead>
<tbody>
<tr>
<td>server.agentId</td>
<td></td>
</tr>
<tr>
<td>server.applications</td>
<td></td>
</tr>
<tr>
<td>server.applications.hasMoreValues</td>
<td></td>
</tr>
<tr>
<td>server.configurationId</td>
<td>✔️</td>
</tr>
<tr>
<td>server.cpuType</td>
<td></td>
</tr>
<tr>
<td>server.hostName</td>
<td></td>
</tr>
<tr>
<td>server.hypervisor</td>
<td></td>
</tr>
<tr>
<td>server.networkInterfaceInfo</td>
<td></td>
</tr>
<tr>
<td>server.networkInterfaceInfo.hasMoreValues</td>
<td></td>
</tr>
</tbody>
</table>

API Version 2015-11-01
### Using the DescribeConfigurationsAction

<table>
<thead>
<tr>
<th>Field</th>
<th>Mandatory</th>
</tr>
</thead>
<tbody>
<tr>
<td>server.osName</td>
<td></td>
</tr>
<tr>
<td>server.osVersion</td>
<td></td>
</tr>
<tr>
<td>server.tags</td>
<td></td>
</tr>
<tr>
<td>server.tags.hasMoreValues</td>
<td></td>
</tr>
<tr>
<td>server.timeOfCreation</td>
<td>✓</td>
</tr>
<tr>
<td>server.type</td>
<td></td>
</tr>
<tr>
<td>server.performance.avgCpuUsagePct</td>
<td></td>
</tr>
<tr>
<td>server.performance.avgDiskReadIOPS</td>
<td></td>
</tr>
<tr>
<td>server.performance.avgDiskReadsPerSecondInKB</td>
<td></td>
</tr>
<tr>
<td>server.performance.avgDiskWriteIOPS</td>
<td></td>
</tr>
<tr>
<td>server.performance.avgDiskWritesPerSecondInKB</td>
<td></td>
</tr>
<tr>
<td>server.performance.avgFreeRAMInKB</td>
<td></td>
</tr>
<tr>
<td>server.performance.avgNetworkReadsPerSecondInKB</td>
<td></td>
</tr>
<tr>
<td>server.performance.avgNetworkWritesPerSecondInKB</td>
<td></td>
</tr>
<tr>
<td>server.performance.maxCpuUsagePct</td>
<td></td>
</tr>
<tr>
<td>server.performance.maxDiskReadIOPS</td>
<td></td>
</tr>
<tr>
<td>server.performance.maxDiskReadsPerSecondInKB</td>
<td></td>
</tr>
<tr>
<td>server.performance.maxDiskWriteIOPS</td>
<td></td>
</tr>
<tr>
<td>server.performance.maxDiskWritesPerSecondInKB</td>
<td></td>
</tr>
<tr>
<td>server.performance.maxNetworkReadsPerSecondInKB</td>
<td></td>
</tr>
<tr>
<td>server.performance.maxNetworkWritesPerSecondInKB</td>
<td></td>
</tr>
<tr>
<td>server.performance.minFreeRAMInKB</td>
<td></td>
</tr>
<tr>
<td>server.performance.numCores</td>
<td></td>
</tr>
<tr>
<td>server.performance.numCpus</td>
<td></td>
</tr>
<tr>
<td>server.performance.numDisks</td>
<td></td>
</tr>
<tr>
<td>server.performance.numNetworkCards</td>
<td></td>
</tr>
<tr>
<td>server.performance.totalRAMInKB</td>
<td></td>
</tr>
</tbody>
</table>

### Process assets

<table>
<thead>
<tr>
<th>Field</th>
<th>Mandatory</th>
</tr>
</thead>
<tbody>
<tr>
<td>process.commandLine</td>
<td></td>
</tr>
</tbody>
</table>
Using the ListConfigurationsAction

The ListConfigurations action 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 ListConfigurations action. The ones marked as mandatory are always present in the output.

Server assets

<table>
<thead>
<tr>
<th>Field</th>
<th>Mandatory</th>
</tr>
</thead>
<tbody>
<tr>
<td>server.configurationId</td>
<td>✔</td>
</tr>
<tr>
<td>server.agentId</td>
<td></td>
</tr>
<tr>
<td>server.hostName</td>
<td></td>
</tr>
<tr>
<td>server.osName</td>
<td></td>
</tr>
<tr>
<td>server.osVersion</td>
<td></td>
</tr>
<tr>
<td>server.timeOfCreation</td>
<td>✔</td>
</tr>
<tr>
<td>server.type</td>
<td></td>
</tr>
</tbody>
</table>

Process assets

<table>
<thead>
<tr>
<th>Field</th>
<th>Mandatory</th>
</tr>
</thead>
<tbody>
<tr>
<td>process.configurationId</td>
<td>✔</td>
</tr>
<tr>
<td>process.name</td>
<td></td>
</tr>
<tr>
<td>process.path</td>
<td></td>
</tr>
<tr>
<td>process.timeOfCreation</td>
<td>✔</td>
</tr>
</tbody>
</table>

Application assets

<table>
<thead>
<tr>
<th>Field</th>
<th>Mandatory</th>
</tr>
</thead>
<tbody>
<tr>
<td>application.configurationId</td>
<td>✔</td>
</tr>
<tr>
<td>application.description</td>
<td></td>
</tr>
<tr>
<td>application.lastModifiedTime</td>
<td>✔</td>
</tr>
<tr>
<td>application.name</td>
<td>✔</td>
</tr>
<tr>
<td>application.serverCount</td>
<td>✔</td>
</tr>
<tr>
<td>application.timeOfCreation</td>
<td>✔</td>
</tr>
<tr>
<td>Field</td>
<td>Mandatory</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>process.commandLine</td>
<td></td>
</tr>
<tr>
<td>process.configurationId</td>
<td>✔</td>
</tr>
<tr>
<td>process.name</td>
<td></td>
</tr>
<tr>
<td>process.path</td>
<td></td>
</tr>
<tr>
<td>process.timeOfCreation</td>
<td>✔</td>
</tr>
<tr>
<td>server.agentId</td>
<td></td>
</tr>
<tr>
<td>server.configurationId</td>
<td>✔</td>
</tr>
</tbody>
</table>

**Application assets**

<table>
<thead>
<tr>
<th>Field</th>
<th>Mandatory</th>
</tr>
</thead>
<tbody>
<tr>
<td>application.configurationId</td>
<td>✔</td>
</tr>
<tr>
<td>application.description</td>
<td></td>
</tr>
<tr>
<td>application.name</td>
<td>✔</td>
</tr>
<tr>
<td>application.serverCount</td>
<td>✔</td>
</tr>
<tr>
<td>application.timeOfCreation</td>
<td>✔</td>
</tr>
<tr>
<td>application.lastModifiedTime</td>
<td>✔</td>
</tr>
</tbody>
</table>

**Connection assets**

<table>
<thead>
<tr>
<th>Field</th>
<th>Mandatory</th>
</tr>
</thead>
<tbody>
<tr>
<td>connection.destinationIp</td>
<td>✔</td>
</tr>
<tr>
<td>connection.destinationPort</td>
<td>✔</td>
</tr>
<tr>
<td>connection.ipVersion</td>
<td>✔</td>
</tr>
<tr>
<td>connection.latestTimestamp</td>
<td>✔</td>
</tr>
<tr>
<td>connection.occurrence</td>
<td>✔</td>
</tr>
<tr>
<td>connection.sourceIp</td>
<td>✔</td>
</tr>
<tr>
<td>connection.transportProtocol</td>
<td></td>
</tr>
<tr>
<td>destinationProcess.configurationId</td>
<td></td>
</tr>
<tr>
<td>destinationProcess.name</td>
<td></td>
</tr>
<tr>
<td>destinationServer.configurationId</td>
<td></td>
</tr>
<tr>
<td>destinationServer.hostName</td>
<td></td>
</tr>
</tbody>
</table>

API Version 2015-11-01
Using the ListConfigurations Action

<table>
<thead>
<tr>
<th>Field</th>
<th>Mandatory</th>
</tr>
</thead>
<tbody>
<tr>
<td>sourceProcess.configurationId</td>
<td></td>
</tr>
<tr>
<td>sourceProcess.name</td>
<td></td>
</tr>
<tr>
<td>sourceServer.configurationId</td>
<td></td>
</tr>
<tr>
<td>sourceServer.hostName</td>
<td></td>
</tr>
</tbody>
</table>

**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**

<table>
<thead>
<tr>
<th>Filter</th>
<th>Supported conditions</th>
<th>Supported values</th>
<th>Supported sorting</th>
</tr>
</thead>
<tbody>
<tr>
<td>server.configurationId</td>
<td>equals</td>
<td>Any valid server configuration ID</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>not_equals</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>eq</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ne</td>
<td></td>
<td></td>
</tr>
<tr>
<td>server.hostName</td>
<td>equals</td>
<td>String</td>
<td>ASC, DESC</td>
</tr>
<tr>
<td></td>
<td>not_equals</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>eq</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ne</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>contains</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>not_contains</td>
<td></td>
<td></td>
</tr>
<tr>
<td>server.osName</td>
<td>equals</td>
<td>String</td>
<td>ASC, DESC</td>
</tr>
<tr>
<td></td>
<td>not_equals</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>eq</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ne</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>contains</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>not_contains</td>
<td></td>
<td></td>
</tr>
<tr>
<td>server.osVersion</td>
<td>equals</td>
<td>String</td>
<td>ASC, DESC</td>
</tr>
<tr>
<td></td>
<td>not_equals</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>eq</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ne</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>contains</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>not_contains</td>
<td></td>
<td></td>
</tr>
<tr>
<td>server.agentId</td>
<td>equals</td>
<td>String</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>not_equals</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>eq</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ne</td>
<td></td>
<td></td>
</tr>
<tr>
<td>server.connectorId</td>
<td>equals</td>
<td>String</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>not_equals</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>eq</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ne</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Filter</td>
<td>Supported conditions</td>
<td>Supported values</td>
<td>Supported sorting</td>
</tr>
<tr>
<td>---------------------</td>
<td>----------------------</td>
<td>----------------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NOT_EQUALS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• EQ</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NE</td>
<td></td>
</tr>
<tr>
<td>server.type</td>
<td>• EQUALS</td>
<td>String with one of the following values:</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>• NOT_EQUALS</td>
<td>• EC2</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• OTHER</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• VMWARE_VM</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• VMWARE_HOST</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• VMWARE_VM_TEMPLATE</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• String</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• CONTAINS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NOT_CONTAINS</td>
<td></td>
</tr>
<tr>
<td>server.vmWareInfo.morefId</td>
<td>• EQUALS</td>
<td>• String</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>• NOT_EQUALS</td>
<td>• CONTAINS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NOT_CONTAINS</td>
<td></td>
</tr>
<tr>
<td>server.vmWareInfo.vcenterId</td>
<td>• EQUALS</td>
<td>• String</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>• NOT_EQUALS</td>
<td>• CONTAINS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NOT_CONTAINS</td>
<td></td>
</tr>
<tr>
<td>server.vmWareInfo.hostId</td>
<td>• EQUALS</td>
<td>• String</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>• NOT_EQUALS</td>
<td>• CONTAINS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NOT_CONTAINS</td>
<td></td>
</tr>
<tr>
<td>server.networkInterfaceInfo.portGroupId</td>
<td>• EQUALS</td>
<td>• String</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>• NOT_EQUALS</td>
<td>• CONTAINS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NOT_CONTAINS</td>
<td></td>
</tr>
<tr>
<td>server.networkInterfaceInfo.portGroupName</td>
<td>• EQUALS</td>
<td>• String</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>• NOT_EQUALS</td>
<td>• CONTAINS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NOT_CONTAINS</td>
<td></td>
</tr>
<tr>
<td>Filter</td>
<td>Supported conditions</td>
<td>Supported values</td>
<td>Supported sorting</td>
</tr>
<tr>
<td>------------------------------</td>
<td>----------------------</td>
<td>------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td><code>server.networkInterfaceInfo.virtualSwitchName</code></td>
<td>• NOT_EQUALS</td>
<td>String</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>• EQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• NE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• NOT_CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>server.networkInterfaceInfo.ipAddress</code></td>
<td>• NOT_EQUALS</td>
<td>String</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>• EQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• NE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• NOT_CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>server.networkInterfaceInfo.macAddress</code></td>
<td>• NOT_EQUALS</td>
<td>String</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>• EQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• NE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• NOT_CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>server.performance.avgCpuUsagePct</code></td>
<td>• GE</td>
<td>Percentage</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>• LE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• GT</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• LT</td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>server.performance.totalDiskFreeSizeInKB</code></td>
<td>• GE</td>
<td>Double</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>• LE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• GT</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• LT</td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>server.performance.avgFreeRAMInKB</code></td>
<td>• GE</td>
<td>Double</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>• LE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• GT</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• LT</td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>server.tag.value</code></td>
<td>• EQUALS</td>
<td>String</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>• NOT_EQUALS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• EQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• NE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• NOT_CONTAINS</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Using the ListConfigurations Action

<table>
<thead>
<tr>
<th>Filter</th>
<th>Supported conditions</th>
<th>Supported values</th>
<th>Supported sorting</th>
</tr>
</thead>
<tbody>
<tr>
<td>server.tag.key</td>
<td>• EQUALS</td>
<td>• String</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>• NOT_EQUALS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• EQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• NE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• NOT_CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>server.application.name</td>
<td>• EQUALS</td>
<td>• String</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>• NOT_EQUALS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• EQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• NE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• NOT_CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>server.application.description</td>
<td>• EQUALS</td>
<td>• String</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>• NOT_EQUALS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• EQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• NE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• NOT_CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>server.application.configurationId</td>
<td>• EQUALS</td>
<td>• Any valid application configuration ID</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>• NOT_EQUALS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• EQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• NE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>server.process.configurationId</td>
<td>• EQUALS</td>
<td>• ProcessId</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>• NOT_EQUALS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• EQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• NE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>server.process.name</td>
<td>• EQUALS</td>
<td>• String</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>• NOT_EQUALS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• EQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• NE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• NOT_CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>server.process.commandLine</td>
<td>• EQUALS</td>
<td>• String</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>• NOT_EQUALS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• EQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• NE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• NOT_CONTAINS</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Application assets
### Application Configuration Filters

<table>
<thead>
<tr>
<th>Filter</th>
<th>Supported conditions</th>
<th>Supported values</th>
<th>Supported sorting</th>
</tr>
</thead>
</table>
| 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 Asset Filters

<table>
<thead>
<tr>
<th>Filter</th>
<th>Supported conditions</th>
<th>Supported values</th>
<th>Supported sorting</th>
</tr>
</thead>
</table>
| process.configurationId | • EQUALS  
• NOT_EQUALS  
• EQ  
• NE | • ProcessId             |                  |
| process.name            | • EQUALS  
• NOT_EQUALS  
• EQ  
• NE  
• CONTAINS  
• NOT_CONTAINS | • String               | • ASC  
• DESC |
### Using the `ListConfigurations` Action

<table>
<thead>
<tr>
<th>Filter</th>
<th>Supported conditions</th>
<th>Supported values</th>
<th>Supported sorting</th>
</tr>
</thead>
<tbody>
<tr>
<td>process.commandLine</td>
<td>• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS</td>
<td>• String</td>
<td>• ASC</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• DESC</td>
</tr>
<tr>
<td>server.configurationId</td>
<td>• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS</td>
<td></td>
<td>• ServerId</td>
</tr>
<tr>
<td>server.hostName</td>
<td>• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS</td>
<td>• String</td>
<td>• ASC</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• DESC</td>
</tr>
<tr>
<td>server.osName</td>
<td>• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS</td>
<td>• String</td>
<td>• ASC</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• DESC</td>
</tr>
<tr>
<td>server.osVersion</td>
<td>• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS</td>
<td>• String</td>
<td>• ASC</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• DESC</td>
</tr>
<tr>
<td>server.agentId</td>
<td>• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS</td>
<td>• String</td>
<td></td>
</tr>
</tbody>
</table>

### Connection assets

<table>
<thead>
<tr>
<th>Filter</th>
<th>Supported conditions</th>
<th>Supported values</th>
<th>Supported sorting</th>
</tr>
</thead>
<tbody>
<tr>
<td>connection.sourceIp</td>
<td>• EQUALS • NOT_EQUALS • EQ</td>
<td>• IP</td>
<td>• ASC</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• DESC</td>
</tr>
</tbody>
</table>
### Using the ListConfigurationsAction

<table>
<thead>
<tr>
<th>Filter</th>
<th>Supported conditions</th>
<th>Supported values</th>
<th>Supported sorting</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>NOT_CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>connection.destinationIp</td>
<td>EQUALS</td>
<td>IP</td>
<td>ASC, DESC</td>
</tr>
<tr>
<td></td>
<td>NOT_EQUALS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>EQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>NE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>NOT_CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>connection.destinationPort</td>
<td>EQUALS</td>
<td>Integer</td>
<td>ASC, DESC</td>
</tr>
<tr>
<td></td>
<td>NOT_EQUALS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>EQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>NE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sourceServer.configurationId</td>
<td>EQUALS</td>
<td>ServerId</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NOT_EQUALS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>EQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>NE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sourceServer.hostName</td>
<td>EQUALS</td>
<td>String</td>
<td>ASC, DESC</td>
</tr>
<tr>
<td></td>
<td>NOT_EQUALS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>EQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>NE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>NOT_CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>destinationServer.osName</td>
<td>EQUALS</td>
<td>String</td>
<td>ASC, DESC</td>
</tr>
<tr>
<td></td>
<td>NOT_EQUALS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>EQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>NE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>NOT_CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>destinationServer.osVersion</td>
<td>EQUALS</td>
<td>String</td>
<td>ASC, DESC</td>
</tr>
<tr>
<td></td>
<td>NOT_EQUALS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>EQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>NE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>NOT_CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>destinationServer.agentId</td>
<td>EQUALS</td>
<td>String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NOT_EQUALS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>EQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>NE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>NOT_CONTAINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Filter</td>
<td>Supported conditions</td>
<td>Supported values</td>
<td>Supported sorting</td>
</tr>
<tr>
<td>--------</td>
<td>----------------------</td>
<td>------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>sourceProcess.configurationId</td>
<td>EQUALS, NOT_EQUALS</td>
<td>• ProcessId</td>
<td></td>
</tr>
<tr>
<td>sourceProcess.name</td>
<td>EQUALS, NOT_EQUALS</td>
<td>• String</td>
<td>ASC, DESC</td>
</tr>
<tr>
<td>sourceProcess.commandLine</td>
<td>EQUALS, NOT_EQUALS</td>
<td>• String</td>
<td>ASC, DESC</td>
</tr>
<tr>
<td>destinationProcess.configurationId</td>
<td>EQUALS, NOT_EQUALS</td>
<td>• ProcessId</td>
<td></td>
</tr>
<tr>
<td>destinationProcess.name</td>
<td>EQUALS, NOT_EQUALS</td>
<td>• String</td>
<td>ASC, DESC</td>
</tr>
<tr>
<td>destinationprocess.commandLine</td>
<td>EQUALS, NOT_EQUALS</td>
<td>• String</td>
<td>ASC, DESC</td>
</tr>
</tbody>
</table>
Actions

The following actions are supported:

- `AssociateConfigurationItemsToApplication` (p. 15)
- `CreateApplication` (p. 17)
- `CreateTags` (p. 20)
- `DeleteApplications` (p. 23)
- `DeleteTags` (p. 25)
- `DescribeAgents` (p. 28)
- `DescribeConfigurations` (p. 32)
- `DescribeContinuousExports` (p. 35)
- `DescribeExportConfigurations` (p. 39)
- `DescribeExportTasks` (p. 42)
- `DescribeTags` (p. 46)
- `DisassociateConfigurationItemsFromApplication` (p. 50)
- `ExportConfigurations` (p. 52)
- `GetDiscoverySummary` (p. 54)
- `ListConfigurations` (p. 57)
- `ListServerNeighbors` (p. 61)
- `StartContinuousExport` (p. 65)
- `StartDataCollectionByAgentIds` (p. 68)
- `StartExportTask` (p. 71)
- `StopContinuousExport` (p. 75)
- `StopDataCollectionByAgentIds` (p. 78)
- `UpdateApplication` (p. 81)
AssociateConfigurationItemsToApplication

Associates one or more configuration items with an application.

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 103).

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. 105).

**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

API Version 2015-11-01
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

Associate configuration items to an application

The following example associates two Application Discovery Service discovered resources specified by their configuration ids to the application identified by the value passed to the parameter applicationConfigurationId in the request.

Sample Request

```json
{
    "applicationConfigurationId": "d-application-0039038d504694533",
    "configurationIds": [
        "d-server-0025db43a885966c8",
        "d-server-0f0a32a0db4217bd6"
    ]
}
```

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. 103).

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. 105).
Example

Create an application

The following example creates an application specified by value passed the required parameter of `name` as well as a description passed to the optional parameter `description` in the request.

Sample Request

```json
{
    "name": "PeopleSoft_phase1",
    "description": "components related to payroll app to be migrated"
}
```

Sample Response

```json
{
    "configurationId": "d-application-0282ccd1ba7c211ca"
}
```

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
See Also

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

Create 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

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 103).

The request accepts the following data in JSON format.

**configurationIds (p. 20)**

A list of configuration items that you want to tag.

Type: Array of strings

Required: Yes

**tags (p. 20)**

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. 101) 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. 105).

**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

Example

Create tags for configuration items

The following example creates two tags, each with a key/value pair, and associates both tags to two Application Discovery Service discovered resources specified by the values passed to the parameter configurationIds in the request.

Sample Request

```json
{
  "configurationIds": [
    "d-server-08a6bce106f6340e",
    "d-server-04d5def51d0a728f3"
  ],
  "tags": [
    {
      "key": "ServerLocation",
      "value": "Warehouse"
    },
    {
      "key": "MigrationPhase",
      "value": "PH1"
    }
  ]
}
```

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

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 103).

The request accepts the following data in JSON format.

`configurationIds` (p. 23)

- 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. 105).

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

Delete applications

The following example deletes two applications as specified by the values passed to the parameter configurationIds in the request.

Sample Request

```json
{
    "configurationIds": [
        "d-application-022b024aa42685464",
        "d-application-0ce38f881a50d2ac1"
    ]
}
```

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

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 103).

The request accepts the following data in JSON format.

configurationIds (p. 25)

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

Type: Array of strings

Required: Yes

tags (p. 25)

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:

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

Type: Array of Tag (p. 101) 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. 105).

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
ServerErrorInternalErrorException

The server experienced an internal error. Try again.

HTTP Status Code: 500

Example

Delete tags

The following example deletes a specific tag identified by its key/value pair from the server as specified by the values passed to the required parameter configurationIds in the request. Since the parameter tags is not required, just passing values to configurationIds would delete all tags associated with the identified server.

Sample Request

```json
{
    "configurationIds": [
    "d-server-08a4bce106f63340e",
    ],
    "tags": [
    {
        "key": "ServerHostType",
        "value": "OracleProdDB"
    }
    ]
}
```

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 connectors as specified by ID or other filters. All agents/connectors associated with your user account can be listed if you call DescribeAgents as is without passing any parameters.

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. 103).

The request accepts the following data in JSON format.

agentIds (p. 28)

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. 28)

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. 98) objects

Required: No

maxResults (p. 28)

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. 28)

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
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. 29)**

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. 85) objects

**nextToken (p. 29)**

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. 105).

**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

Describe agents that are running

The following example describes all agents that are currently running by passing object array values to the parameter filters where condition is "EQUALS", name is "health", and value is "RUNNING". This results in a descriptive response that lists all agents running in your account.

Sample Request

```
{
  "filters": [
    {
      "condition": "EQUALS",
      "name": "health",
      "values": [ "RUNNING" ]
    }
  ]
}
```

Sample Response

```
{
  "agentsInfo": [
    {
      "agentId": "o-55vu6xeo8zs27b905",
      "hostName": "ip-172-31-24-158.us-west-2.compute.internal",
      "agentNetworkInfoList": [
        {
          "ipAddress": "172.31.24.158",
          "macAddress": "02:7C:B6:F7:82:C4"
        }
      ],
      "version": "2.0.1008.0",
      "health": "RUNNING",
      "lastHealthPingTime": "2018-03-19T22:48:25Z",
      "collectionStatus": "STARTED",
      "agentType": "ONPREMISES",
    }
  ]
}
```
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.

**Note**

All of the supplied IDs must be for the same asset type from one of the following:

- server
- application
- process
- connection

Output fields are specific to the asset type specified. For example, the output for a server configuration item includes a list of attributes about the server, such as host name, operating system, number of network cards, etc.

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

**Request Syntax**

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

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 103).

The request accepts the following data in JSON format.

**configurationIds (p. 32)**

- One or more configuration IDs.
- Type: Array of strings
- Required: Yes

**Response Syntax**

```json
{
   "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. 32)

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. 105).

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

Describe configurations

The following example queries configuration descriptions of two applications as specified by the values passed to the parameter configurationIds in the request. Note that the configuration ID's passed meet the requirement that they represent the same asset type, in this example, application.

Sample Request

```json
{
  "configurationIds": [
    "d-application-02822cdd1ba7c211ca",
    "d-application-034219054305e4a34"
  ]
}
```

Sample Response

```json
{

```
"configurations": [
{
   "application.configurationId": "d-application-0282ccd1ba7c211ca",
   "application.description": "ads CreateApplication",
   "application.lastModifiedTime": "2018-02-27 00:48:16.0",
   "application.name": "adsdp2",
   "application.serverCount": "2",
   "application.timeOfCreation": "2018-02-27 00:48:16.0"
},
{
   "application.configurationId": "d-application-034219054305e4a34",
   "application.description": "ads CreateApplication place holder app",
   "application.lastModifiedTime": "2018-02-27 23:33:32.0",
   "application.name": "adsdp2_ph1",
   "application.serverCount": "0",
   "application.timeOfCreation": "2018-02-27 23:33:32.0"
}
]

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
DescribeContinuousExports

Lists exports as specified by ID. All continuous exports associated with your user account can be listed if you call DescribeContinuousExports as is without passing any parameters.

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. 103).

The request accepts the following data in JSON format.

exportIds (p. 35)

The unique IDs assigned to the exports.

Type: Array of strings

Required: No

maxResults (p. 35)

A number between 1 and 100 specifying the maximum number of continuous export descriptions returned.

Type: Integer

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

Required: No

nextToken (p. 35)

The token from the previous call to DescribeExportTasks.

Type: String

Required: No

Response Syntax

```
{
    "descriptions": [
        {
            "dataSource": "string",
            "exportId": "string",
            "s3Bucket": "string",
            "schemaStorageConfig": {
                "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.

descriptions (p. 35)

A list of continuous export descriptions.

Type: Array of ContinuousExportDescription (p. 89) objects

nextToken (p. 35)

The token from the previous call to DescribeExportTasks.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 105).

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

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

Example

Describe a continuous export

The following example shows the request and response of describing a particular continuous export specified by passing in a value to exportId as well as other filtering parameters.

Sample Request

```
{
  "maxResults": 10,
  "nextToken": "rLH9RloHsWY1sljKfW1mHioorH1JjJ1TC9MD06H0lpINQg8QFk302u5n9YF59JZ5k5NzAgs6YerZWy",
  "exportIds": [
    "continuous-export-181b77c3-7jf3-4610-924e-7acafe5bbc59"
  ]
}
```

Sample Response

```
{
  "descriptions": [ 
    {
      "exportId": "continuous-export-181b77c3-7jf3-4610-924e-7acafe5bbc59",
      "status": "ACTIVE"
    },
    {
      "schemaStorageConfig": {
        "databaseName": "application_discovery_service_database"
      },
      "startTime": 1531248144.117,
      "exportId": "continuous-export-181b77c3-7jf3-4610-924e-7acafe5bbc59",
      "s3Bucket": "aws-application-discovery-service-92costafer5c9vn0gjane94mi",
      "status": "STOP_IN_PROGRESS",
      "stopTime": 1531248232.085,
      "statusDetail": ",",
      "dataSource": "AGENT"
    }
  ]
}
```

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++
See Also

- 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

DescribeExportConfigurations is deprecated.

Use instead DescribeExportTasks.

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 103).

The request accepts the following data in JSON format.

**exportIds (p. 39)**

A list of continuous export ids to search for.

Type: Array of strings

Required: No

**maxResults (p. 39)**

A number between 1 and 100 specifying the maximum number of continuous export descriptions returned.

Type: Integer

Required: No

**nextToken (p. 39)**

The token from the previous call to describe-export-tasks.

Type: String

Required: No

Response Syntax

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

**exportsInfo (p. 39)**

Type: Array of ExportInfo (p. 96) objects

**nextToken (p. 39)**

The token from the previous call to describe-export-tasks.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 105).

**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:
See Also

• 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

```json
{
    "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. 103).

The request accepts the following data in JSON format.

**exportIds (p. 42)**

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

Type: Array of strings

Required: No

**filters (p. 42)**

One or more filters.

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

Type: Array of ExportFilter (p. 95) objects

Required: No

**maxResults (p. 42)**

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. 42)**

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. 43)**

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. 96) objects

**nextToken (p. 43)**

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. 105).

**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",
      "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": 
"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 have tags as specified by the key-value pairs, name and value, passed to the optional parameter filters.

There are three valid tag filter names:

• tagKey
• tagValue
• configurationId

Also, all configuration items associated with your user account that have tags can be listed if you call DescribeTags as is without passing any parameters.

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. 103).

The request accepts the following data in JSON format.

filters (p. 46)

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. 102) objects

Required: No

maxResults (p. 46)

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

Type: Integer

Required: No

nextToken (p. 46)

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. 47)**

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

Type: String

**tags (p. 47)**

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. 88) objects

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 105).

**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

**Example**

**Describe tags that have a particular value**

The following example lists configuration items associated with an agent installed on a linux server. This is done by passing object array values to the parameter filters where *name* is "tagValue" and *value* is "agent on linux server". This results in a descriptive response that lists configuration item details where the tag value is equal to the value specified in the request.

**Sample Request**

```json
{
  "filters": [
    {
      "name": "tagValue",
      "values": [ "agent on linux server" ]
    }
  ]
}
```

**Sample Response**

```json
{
  "tags": [
    {
      "configurationType": "SERVER",
      "configurationId": "d-server-0c269e353e6d1fb4c",
      "key": "dp2",
      "value": "agent on linux server",
      "timeOfCreation": 1519674610.0
    },
    {
      "nextToken": "0039038d504694533"
    }
  ],
}
```

**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
See Also

- 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

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 103).

The request accepts the following data in JSON format.

applicationConfigurationId (p. 50)

- Configuration ID of an application from which each item is disassociated.
- Type: String
- Required: Yes

configurationIds (p. 50)

- 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. 105).

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

ServerErrorInternalErrorException

The server experienced an internal error. Try again.

HTTP Status Code: 500

Example

Disassociate configuration items from an application

The following example disassociates two Application Discovery Service discovered resources specified by their configuration ids from the application identified by the value passed to the required parameter applicationConfigurationId in the request.

Sample Request

```json
{
   "applicationConfigurationId": "d-application-0039038d504694533",
   "configurationIds": [
      "d-server-0025db43a885966c8",
      "d-server-0f0a32a0db4217bd6"
   ]
}
```

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. 52)

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. 105).

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.

This API operation takes no request parameters and is called as is at the command prompt as shown in the example.

Response Syntax

```json
{
  "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. 54)

Details about discovered agents, including agent status and health.

Type: CustomerAgentInfo (p. 91) object

applications (p. 54)

The number of applications discovered.

Type: Long

connectorSummary (p. 54)

Details about discovered connectors, including connector status and health.

Type: CustomerConnectorInfo (p. 93) object

servers (p. 54)

The number of servers discovered.
Type: Long
serversMappedToApplications (p. 54)
The number of servers mapped to applications.

Type: Long
serversMappedToTags (p. 54)
The number of servers mapped to tags.

Errors
For information about the errors that are common to all actions, see Common Errors (p. 105).

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
Get discovery summary of discovered assets
In the following example GetDiscoverySummary is called as is, without any request parameters, directly from the AWS CLI and returns a summary of all discovered assets associated with your account.

Sample Request

```
[1212|ttys000|510]$ aws discovery get-discovery-summary
```

Sample Response
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 as specified by the value passed to the required parameter configurationType. Optional filtering may be applied to refine search results.

Request Syntax

```json
{
  "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. 103).

The request accepts the following data in JSON format.

**configurationType (p. 57)**

A valid configuration identified by Application Discovery Service.

Type: String

Valid Values: SERVER | PROCESS | CONNECTION | APPLICATION

Required: Yes

**filters (p. 57)**

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

```json
{"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. 98) objects

Required: No

**maxResults (p. 57)**

The total number of items to return. The maximum value is 100.
Type: Integer  
Required: No  

nextToken (p. 57)  
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. 57)  
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. 100) objects  
Required: No  

Response Syntax

```json
{
  "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. 58)  
Returns configuration details, including the configuration ID, attribute names, and attribute values.

Type: Array of string to string maps

nextToken (p. 58)  
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. 105).
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

ServerErrorInternalErrorException

The server experienced an internal error. Try again.

HTTP Status Code: 500

Example

List configurations based on server type

The following example lists server configurations where the server type is an EC2 instance. This is done by passing the value "SERVER" to the required parameter configurationType. Then, to limit the results to just EC2 servers, object array values are passed to the parameter filters where condition is "EQUALS", name is "server.type", and values is "EC2". A sorting refinement is also used in this example by passing values to the parameter orderBy that sorts the results by the server's OS name in reverse alphabetical order.

Sample Request

```json
{
    "configurationType": "SERVER",
    "filters": [
        {
            "condition": "EQUALS",
            "name": "server.type",
            "values": [ "EC2" ]
        }
    ],
    "orderBy": [
        {
            "fieldName": "server.osName",
            "sortOrder": "DESC"
        }
    ]
}
```
Sample Response

```json
{
  "configurations": [
    {
      "server.agentId": "i-0adf772ed9f61445a",
      "server.configurationId": "d-server-062d1f42166d190a4",
      "server.hostName": "WIN-7092JH2AKNB",
      "server.osName": "WindowsOS - Windows Server 2012 R2 x64",
      "server.osVersion": "6.2.9200",
      "server.source": "Agent",
      "server.timeOfCreation": "2017-10-09 20:42:25.0",
      "server.type": "EC2"
    },
    {
      "server.agentId": "i-0243426187d59c9c3",
      "server.configurationId": "d-server-0fc7f9126ed1c2be7",
      "server.hostName": "ip-10-0-0-22.us-west-2.compute.internal",
      "server.osName": "Linux - Red Hat Enterprise Linux Server release 7.3 (Maipo)",
      "server.osVersion": "3.10.0-514.el7.x86_64",
      "server.source": "Agent",
      "server.timeOfCreation": "2017-04-06 19:58:15.0",
      "server.type": "EC2"
    }
  ]
}
```

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. 103).

The request accepts the following data in JSON format.

`configurationId (p. 61)`

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

Type: String

Required: Yes

`maxResults (p. 61)`

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

Type: Integer

Required: No

`neighborConfigurationIds (p. 61)`

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

Type: Array of strings

Required: No

`nextToken (p. 61)`

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. 61)`

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

Type: Boolean
Response Syntax

```json
{
    "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. 62)**

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

Type: Long

**neighbors (p. 62)**

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

Type: Array of NeighborConnectionDetail (p. 99) objects

**nextToken (p. 62)**

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. 105).

**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

List server neighbors

The following example requests to find out if there are any server neighbors that are one network hop away from the server specified in the required parameter `configurationId`. Port information is also requested by passing "true" to the boolean parameter `portInformationNeeded` (note that boolean parameters must be passed as all lowercase without any quotes). In this particular example, no server neighbors are found that are one network hop away.

**Sample Request**

```json
{
  "configurationId": "d-server-0974b967a76335589",
  "neighborConfigurationIds": [ "d-server-0c269e353e6d1fb4c","d-server-01440c9c5df26ae50" ],
  "portInformationNeeded": true
}
```

**Sample Response**

```json
{
  "neighbors": [],
  "knownDependencyCount": 0
}
```

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
**StartContinuousExport**

Start the continuous flow of agent's discovered data into Amazon Athena.

**Response Syntax**

```
{
  "dataSource": "string",
  "exportId": "string",
  "s3Bucket": "string",
  "schemaStorageConfig": {
    "string": "string"
  },
  "startTime": 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.

**dataSource (p. 65)**

The type of data collector used to gather this data (currently only offered for AGENT).

Type: String

Valid Values: AGENT

**exportId (p. 65)**

The unique ID assigned to this export.

Type: String

**s3Bucket (p. 65)**

The name of the s3 bucket where the export data parquet files are stored.

Type: String

**schemaStorageConfig (p. 65)**

A dictionary which describes how the data is stored.

- **databaseName** - the name of the Glue database used to store the schema.

  Type: String to string map

  Key Length Constraints: Minimum length of 1. Maximum length of 252.

**startTime (p. 65)**

The timestamp representing when the continuous export was started.

Type: Timestamp

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 105).
**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

**ConflictErrorException**

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

**ResourceInUseException**

HTTP Status Code: 400

**ServerInternalErrorException**

The server experienced an internal error. Try again.

HTTP Status Code: 500

---

**Example**

**Start a continuous export**

The following example shows the response of starting a continuous export.

**Sample Response**

```
{
    "startTime": 1531247730.19,
    "dataSource": "AGENT",
    "schemaStorageConfig": {
        "databaseName": "application_discovery_service_database"
    },
    "exportId": "continuous-export-181b77c3-7jf3-4610-924e-7acafe5bb59",
    "s3Bucket": "aws-application-discovery-service-92costafer5c9vn0gjane94mi"
}
```

**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

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 103).

The request accepts the following data in JSON format.

agentIds (p. 68)

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

```json
{
    "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. 68)

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.
Errors

For information about the errors that are common to all actions, see Common Errors (p. 105).

**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

Start data collection by agent ids for one connector and one agent

In the following example, multiple agent ids are passed to the required parameter `agentIds` to start one connector and one agent. In the response, the first agent id returned is reported as started successfully, in the second agent id returned, the description reads "Redundant" which means that a request was sent to start an agent that was already running.

**Sample Request**

```json
{
  "agentIds": [ "c-a07gapohc6spvdl65", "i-0afec73de85777dd0" ]
}
```

**Sample Response**

```json
{
  "agentsConfigurationStatus": [
    {
      "agentId": "c-a07gapohc6spvdl65",
      "operationSucceeded": true,
    
```
"description": "Succeeded"
},
{
   "agentId": "i-0afec73de85777dd0",
   "operationSucceeded": false,
   "description": "Redundant"
}
]

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

Starts 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

```json
{
  "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. 103).

The request accepts the following data in JSON format.

**endTime (p. 71)**

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. 71)**

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. 71)**

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. 95) objects

Required: No

startTime (p. 71)

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. 72)

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. 105).

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
**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",
        "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
```
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
StopContinuousExport

Stop the continuous flow of agent's discovered data into Amazon Athena.

**Request Syntax**

```json
{
    "exportId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 103).

The request accepts the following data in JSON format.

**exportId (p. 75)**

The unique ID assigned to this export.

Type: String

Required: Yes

**Response Syntax**

```json
{
    "startTime": number,
    "stopTime": 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.

**startTime (p. 75)**

Timestamp that represents when this continuous export started collecting data.

Type: Timestamp

**stopTime (p. 75)**

Timestamp that represents when this continuous export was stopped.

Type: Timestamp

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 105).
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

ResourceInUseException

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

Example

Stop a continuous export.

The following example shows the request and response of stopping a particular continuous export specified by passing in a value to exportId.

Sample Request

```json
{
    "exportId": "continuous-export-181b77c3-7jf3-4610-924e-7acafe5bbc59"
}
```

Sample Response

```json
{
    "ApplicationStatus": "IN_PROGRESS",
}
```
"LastUpdatedTime": 1493405005.639
}

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**

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

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 103).

The request accepts the following data in JSON format.

**agentIds (p. 78)**

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

Type: Array of strings

Required: Yes

**Response Syntax**

```json
{
  "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. 78)**

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. 84) objects

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 105).
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

Stop data collection by agent ids for two agents

In the following example, multiple agent ids are passed to the required parameter `agentIds` to stop two agents. In the response, it is reported that both agents were stopped successfully.

Sample Request

```json
{
    "agentIds": [ "i-0afec73de85777dd0","i-0b30b932b328cbd73"
}
```

Sample Response

```json
{
    "agentsConfigurationStatus": [ 
        {
            "agentId": "i-0afec73de85777dd0",
            "operationSucceeded": true,
            "description": "Succeeded"
        },
        {
            "agentId": "i-0b30b932b328cbd73",
            "operationSucceeded": true,
            "description": "Succeeded"
        }
    ]
}
```

See Also

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

- 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

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 103).

The request accepts the following data in JSON format.

configurationId (p. 81)

  Configuration ID of the application to be updated.

  Type: String

  Required: Yes

description (p. 81)

  New description of the application to be updated.

  Type: String

  Required: No

name (p. 81)

  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. 105).

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
Example

Update application name and description

In the following example, both the name and the description are changed (updated) of an existing application. The application is identified by passing the application id to the required parameter configurationId.

Sample Request

```
{
    "configurationId": "d-application-03767f7bddd6c0531",
    "description": "PSoft financials db migration",
    "name": "payroll_db_migration"
}
```

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 (p. 84)
- AgentInfo (p. 85)
- AgentNetworkInfo (p. 87)
- ConfigurationTag (p. 88)
- ContinuousExportDescription (p. 89)
- CustomerAgentInfo (p. 91)
- CustomerConnectorInfo (p. 93)
- ExportFilter (p. 95)
- ExportInfo (p. 96)
- Filter (p. 98)
- NeighborConnectionDetail (p. 99)
- OrderByElement (p. 100)
- Tag (p. 101)
- TagFilter (p. 102)
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. 87) 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
ContinuousExportDescription

A list of continuous export descriptions.

Contents

datasource

The type of data collector used to gather this data (currently only offered for AGENT).

Type: String

Valid Values: AGENT

Required: No

exportId

The unique ID assigned to this export.

Type: String

Required: No

s3Bucket

The name of the s3 bucket where the export data parquet files are stored.

Type: String

Required: No

schemaStorageConfig

An object which describes how the data is stored.

- `databaseName` - the name of the Glue database used to store the schema.

Type: String to string map

Key Length Constraints: Minimum length of 1. Maximum length of 252.

Required: No

startTime

The timestamp representing when the continuous export was started.

Type: Timestamp

Required: No

status

Describes the status of the export. Can be one of the following values:

- START_IN_PROGRESS - setting up resources to start continuous export.
- START_FAILED - an error occurred setting up continuous export. To recover, call start-continuous-export again.
- ACTIVE - data is being exported to the customer bucket.
- ERROR - an error occurred during export. To fix the issue, call stop-continuous-export and start-continuous-export.
- STOP_IN_PROGRESS - stopping the export.
- **STOP_FAILED** - an error occurred stopping the export. To recover, call stop-continuous-export again.
- **INACTIVE** - the continuous export has been stopped. Data is no longer being exported to the customer bucket.

  Type: String
  Valid Values: START_IN_PROGRESS | START_FAILED | ACTIVE | ERROR | STOP_IN_PROGRESS | STOP_FAILED | INACTIVE
  Required: No

### statusDetail

Contains information about any errors that may have occurred.

Type: String
Length Constraints: Minimum length of 1. Maximum length of 255.
Required: No

### stopTime

The timestamp that represents when this continuous export was stopped.

Type: Timestamp
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

collectionsCount
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