

# HP Operations Orchestration Software

Software Version: 9.00.03

## *Amazon Elastic Compute Cloud Integration Guide*

Document Release Date: November 2010

Software Release Date: November 2010



## Legal Notices

### Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

### Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

### Copyright Notices

© Copyright 2010 Hewlett-Packard Development Company, L.P.

### Trademark Notices

For information on open-source and third-party software acknowledgements, see in the documentation set for this release, Open-Source and Third-Party Software Acknowledgements (3rdPartyOpenNotices.pdf).

# On the Web: Finding OO support and documentation

There are two Web sites where you can find support and documentation, including updates to OO Help systems, guides, and tutorials:

- The OO Support site
- HP Live Network

## Support

Documentation enhancements are a continual project at Hewlett-Packard Software. You can obtain or update the HP OO documentation set and tutorials at any time from the HP Software Product Manuals Web site. You will need an HP Passport to log in to the Web site.

### To obtain HP OO documentation and tutorials

1. Go to the HP Software Product Manuals Web site (<http://support.openview.hp.com/selfsolve/manuals>).
2. Log in with your HP Passport user name and password.

OR

If you do not have an HP Passport, click **New users – please register** to create an HP Passport, then return to this page and log in.

If you need help getting an HP Passport, see your HP OO contact.

3. In the **Product** list box, scroll down to and select **Operations Orchestration**.
4. In the **Product Version** list, click the version of the manuals that you're interested in.
5. In the **Operating System** list, click the relevant operating system.
6. Click the **Search** button.
7. In the **Results** list, click the link for the file that you want.

## HP Live Network

For support information, including patches, troubleshooting aids, support contract management, product manuals and more, visit the following site: <https://www.www2.hp.com/>.

This is the **HP Live Network** Web page. To sign in:

1. Click **Login**.
2. On the **HP Passport sign-in** page, enter your HP Passport user ID and password and then click **Sign-in**.
3. If you do not already have an HP Passport account, do the following:
  - a. On the **HP Passport sign-in** page, click **New user registration**.
  - b. On the **HP Passport new user registration** page, enter the required information and then click **Continue**.
  - c. On the confirmation page that opens, check your information and then click **Register**.
  - d. On the **Terms of Service** page, read the Terms of use and legal restrictions, select the **Agree** button, and then click **Submit**.
4. On the **HP Live Network** page, click **Operations Orchestration Community**.

**The Operations Orchestration Community** page contains links to announcements, discussions, downloads, documentation, help, and support.

**Note:** Contact your OO contact if you have any difficulties with this process.

## In OO: How to find Help, PDFs, and tutorials

The HP Operations Orchestration software (HP OO) documentation set is made up of the following:

- Help for Central

Central Help provides information to the following:

- Finding and running flows
- For HP OO administrators, configuring the functioning of HP OO
- Generating and viewing the information available from the outcomes of flow runs

The Central Help system is also available as a PDF document in the HP OO home directory, in the \Central\docs subdirectory.

- Help for Studio

Studio Help instructs flow authors at varying levels of programming ability.

The Studio Help system is also available as a PDF document in the HP OO home directory, in the \Studio\docs subdirectory.

- Animated tutorials for Central and Studio

HP OO tutorials can each be completed in less than half an hour and provide basic instruction on the following:

- In Central, finding, running, and viewing information from flows
- In Studio, modifying flows

The tutorials are available in the Central and Studio subdirectories of the HP OO home directory.

- Self-documentation for operations and flows in the Accelerator Packs and ITIL folders

Self-documentation is available in the descriptions of the operations and steps that are included in the flows.

# Table of Contents

Warranty .....	ii
Restricted Rights Legend .....	ii
Trademark Notices .....	ii
On the Web: Finding OO support and documentation.....	iii
Support .....	iii
HP Live Network .....	iii
In OO: How to find Help, PDFs, and tutorials.....	iv
Overview of Amazon Elastic Compute Cloud integration .....	1
Use cases and scenarios .....	1
Installation and configuration instructions.....	2
Versions .....	3
Architecture .....	3
EC2 integration operation and flow infrastructure.....	3
Common inputs in the integration .....	4
Operation and flow specifics .....	5
Availability zones and regions .....	5
Enumerate Availability Zones .....	5
Enumerate Regions .....	6
Elastic block store .....	6
Snapshots .....	6
Volumes.....	11

Elastic IPs .....	16
Allocate New Elastic Ip .....	16
Associate Elastic Ip .....	16
Disassociate Elastic Ip .....	16
Enumerate Elastic Ips .....	17
Release Elastic Ip .....	17
Images .....	18
Enumerate Images.....	18
Get Image Details .....	19
Modify Image Attribute.....	20
Register Image.....	21
Remove Image.....	22
Reset Image Attribute .....	22
Instances.....	23
Describe Instances.....	23
Get Instance Details.....	23
Get System Log.....	25
Monitor Instance .....	25
Restart Instance.....	26
Run Instances .....	26
Run Instances From Template .....	28
Start Instance.....	29
Stop Instance .....	30
Terminate Instance.....	30
Unmonitor Instance.....	31
Key Pairs .....	31
Allocate New Key Pair.....	31
Delete Key Pair .....	32
Enumerate Key Pairs.....	32
Security groups .....	33
Allow Access To Security Group .....	33
Create Security Group .....	33
Delete Security Group .....	34
Describe Security Groups .....	34
Get Security Group Details .....	35
Revoke Access To Security Group.....	36
Windows.....	37
Bundle Windows Instance .....	37
Cancel Bundle Task.....	38
Enumerate Bundle Tasks .....	39
Get Bundle Task Details.....	39

EC2 Dashboard .....	40
<b>Troubleshooting .....</b>	<b>40</b>
General troubleshooting procedures and tools .....	40
Error messages .....	41
<b>Security .....</b>	<b>42</b>
<b>Tools .....</b>	<b>43</b>

# Overview of Amazon Elastic Compute Cloud integration

With this integration, you can build HP Operations Orchestration (OO) flows that are integrated into the Amazon Elastic Compute Cloud (EC2).

The EC2 integration uses the EC2 Query API released on 11/30/2009 to integrate with OO. To use this integration successfully, you should have knowledge of the EC2 technology.

This document will explain how this integration has been implemented and how the operations included communicate between OO and EC2.

## Use cases and scenarios

The following are the major use cases for the Amazon EC2 integration, and the operations and flows that you can use to implement them.

### 1. Enumerate availability zones and regions:

- Enumerate Availability Zones
- Enumerate Regions

### 2. Manage elastic block store:

- Snapshots
  - Create Shared Snapshot
  - Create Snapshot
  - Delete Snapshot
  - Describe Snapshots
  - Get Snapshot Details
  - Modify Snapshot Attribute
  - Reset Snapshot Attribute
- Volumes
  - Attach Volume
  - Create Volume
  - Delete Volume
  - Describe Volumes
  - Detach Volume
  - Get Volumes Details

### 3. Work with elastic IPs:

- Allocate New Elastic Ip
- Associate Elastic Ip
- Disassociate Elastic Ip
- Enumerate Elastic Ips
- Release Elastic Ip

### 4. Manage images:

- Enumerate Images
- Get Image Details
- Modify Image Attribute
- Register Image

- Remove Image
  - Reset Image Attribute
5. Manage instances:
    - Describe Instances
    - Get Instance Details
    - Get System Log
    - Monitor Instance
    - Restart Instance
    - Run Instances
    - Run Instances From Template
    - Start Instance
    - Stop Instance
    - Terminate Instance
    - Unmonitor Instance
  6. Manage key pairs:
    - Allocate New Key Pair
    - Delete Key Pair
    - Enumerate Key Pairs
  7. Manage security groups:
    - Allow Access To Security Group
    - Create Security Group
    - Delete Security Group
    - Describe Security Groups
    - Get Security Group Details
    - Revoke Access To Security Group
  8. Manage Windows bundle tasks:
    - Bundle Windows Instance
    - Cancel Bundle Task
    - Enumerate Bundle Tasks
    - Get Bundle Task Details
  9. Manage EC2 Dashboard

## Installation and configuration instructions

No special installation and configuration instructions are required for the Amazon EC2 integration. To access the Amazon EC2 console, go to <http://aws.amazon.com/console/> and use your EC2 credentials to log on. Amazon has created specialized plugins for several browsers, such as Elasticfox for Firefox, that allow easy interaction with Amazon EC2.

# Versions

Operations Orchestration Version	Amazon EC2 Version
9.00.03	EC2 API Version 2009-11-30

# Architecture

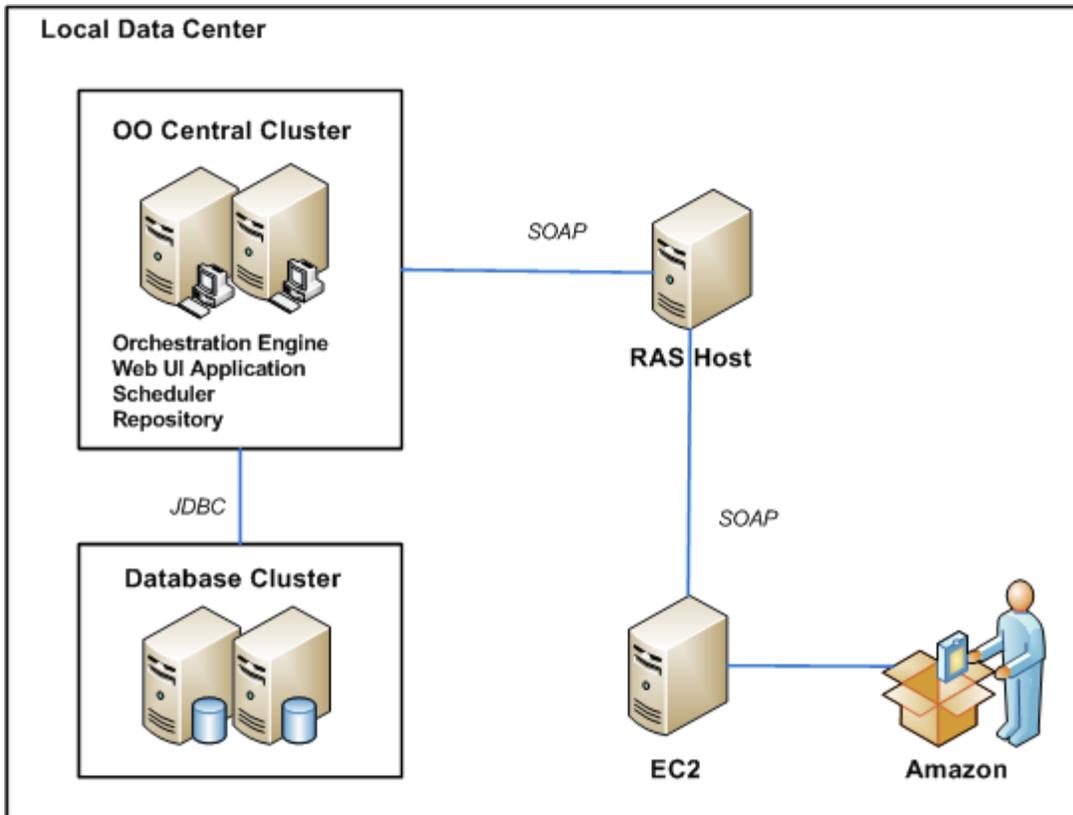


Figure 1 - Amazon EC2 architecture

# EC2 integration operation and flow infrastructure

The EC2 integration includes the following operations in the OO Studio Library/Integrations/Amazon/EC2/ folder.



Figure 2 – Amazon EC2 integration operation and flow infrastructure

## Common inputs in the integration

OO flows and operations use inputs to specify how they obtain the data that they need and when the data is obtained. The following inputs are used consistently throughout the EC2 integration's operations and flows.

### **accessKeyId**

The ID of the access key that uniquely identifies the user account. This can be obtained from the Amazon console.

**accessKey**

The secret access key that uniquely identifies the user. This can be obtained from the Amazon console.

**proxyHost**

The proxy host used for accessing the Internet. This is an optional input.

**proxyPort**

The port used for the proxy. This is an optional input.

**serviceEndpoint**

The endpoint to which the requests are sent. This is an optional input. The default is **us-east-1.ec2.amazonaws.com**.

## Operation and flow specifics

This section describes the Amazon EC2 integration's operations and flows, including any operation- or flow-specific inputs. The flows and operations are grouped by their basic functionality:

- Availability zones and regions
- Elastic block store—snapshots and volumes
- Elastic IPs
- Images
- Instances
- Key pairs
- Security groups
- Windows bundle tasks
- EC2 Dashboard

The sample flows in the OO Library/Integrations/Amazon/EC2/ folder perform some of the most common tasks that need to be automated when using EC2, such as starting instances, allocating elastic IPs, and working with images. Each of these sample flows has a description that describes in detail what it does. You can use these flows as they are or as templates for new operations.

### Availability zones and regions

#### Enumerate Availability Zones

The **Enumerate Availability Zones** operation retrieves a list of your Amazon EC2 availability zones.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**delimiter**

The delimiter used for the results list. The default is a comma (,).

The operation returns the following:

**requestId**

The ID of the request you have sent to Amazon by using this operation.

## returnResult

A delimiter-separated list of your available availability zones.

## notAvailableZones

A delimiter-separated list of availability zones which are not available for use, but are visible to you (this list usually has no elements).

## Notes:

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.
- Availability zones are not the same across accounts. The availability zone **us-east-1a** for account A is not necessarily the same as **us-east-1a** for account B.

## Enumerate Regions

The **Enumerate Regions** operation lists the regions that are currently available to you.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

## rowDelimiter

The delimiter to place between the rows of the results table. The default is the line separator character for the current operating system.

## colDelimiter

The delimiter to be placed between the columns of the results table. The default is a comma (,).

The operation returns the following:

## requestId

The ID of the request you have sent to Amazon by using this operation.

## returnResult

The available regions in a table having the following columns: **Region name** and **Region endpoint**.

### Example:

[eu-west-1](#),[eu-west-1.ec2.amazonaws.com](#)

[us-east-1](#),[us-east-1.ec2.amazonaws.com](#)

## Notes:

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

## Elastic block store

### Snapshots

#### Create Shared Snapshot

The **Create Shared Snapshot** flow creates a snapshot of an EBS volume and makes the snapshot public or grants permission to it for a given user. The flow returns the ID of the generated snapshot. If you enter a value of **all** for the **groups** input, the snapshot is shared publicly.

Otherwise, the snapshot is private and only the developers whose account numbers are supplied through the **userIds** input are granted access to it.

All of the flow's inputs except the following are described in [Common inputs in the integration](#).

**volumeId**

The ID of the volume from which the snapshot is generated.

**userIds**

A list of the users to whom you grant access. This list can only contain IDs, which must be numeric.

**groups**

A list of the groups to which you grant access (according to Amazon, currently the only supported value is **all**; other values may be supported in the future).

**delimiter**

The delimiter for the elements of the inputs that contain lists.

The flow returns the following:

**returnResult**

The ID of the generated snapshot.

**Notes:**

- At least one of the **userIds** or **groups** inputs should not be empty.
- Like all Amazon EC2 operations, the results might not be visible instantly. Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

### **Create Snapshot**

The **Create Snapshot** operation creates a snapshot of an EBS volume and stores it in S3. It returns the ID of the generated snapshot.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**volumeId**

The ID of the volume from which the snapshot is generated.

**snapshotDescription**

The description of the snapshot to be created.

The operation also returns the following:

**returnCode**

The ID of the generated snapshot.

**requestId**

The Amazon request ID.

**volumeId**

The ID of the volume from which the snapshot is generated.

**status**

The snapshot state.

**startTime**

The timestamp when the snapshot was initiated.

**progress**

The progress of the snapshot, in percentage.

**ownerId**

The ID of the snapshot's owner.

**volumeSize**

The size of the volume from which the snapshot is generated.

**description**

The description of the snapshot.

**Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

### **Delete Snapshot**

The **Delete Snapshot** operation deletes a snapshot of an Amazon EBS volume. It returns **true** on success and **false** on failure.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**snapshotId**

The ID of the snapshot to be deleted.

The operation returns the following:

**returnCode**

**true** on success and **false** on failure.

**requestId**

The Amazon request ID.

**Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

### **Describe Snapshots**

The **Describe Snapshots** operation describes the status of Amazon EBS snapshots. If you provide a list of snapshot IDs, only those snapshots are described. Otherwise, all of the existing snapshots are described.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**snapshotIds**

An optional list of snapshot IDs. If you provide this list, only these snapshots are described.

**delimiter**

The delimiter to place between the elements of the **snapshotIds**, **snapshotOwners**, and **restorableBy** inputs. The default is a comma (,).

**rowDelimiter**

The delimiter to place between the rows of an output table. The default is the line separator of the current operating system.

**colDelimiter**

The delimiter to place between the columns of an output table. The default is a comma (,).

**snapshotOwners**

The owner of the snapshots to return. You can specify multiple owners using the delimiter specified in the **delimiter** input. The valid values are **self**, **amazon**, and **AWS Account IDs**.

**restorableBy**

A list of restorable users separated by the delimiter specified in the **delimiter** input. If you specify values for this input, only users who have create snapshot permissions for the snapshots are returned.

The operation returns the following:

**returnResult**

A table containing the following columns: the snapshot ID, the ID of the volume from which the snapshot was created, the volume size, the snapshot status (**pending**, **completed**, or **error**), the timestamp when the snapshot was initiated, and the progress of the snapshot creation process, in percentage.

Each row in the table corresponds to a snapshot. If a column's result is empty, 'null' is returned instead.

**requestId**

The Amazon request ID.

**Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

### ***Get Snapshot Details***

The **Get Snapshot Details** operation describes the status of an EBS snapshot.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**snapshotId**

The ID of the snapshot to be described.

The operation returns the following:

**requestId**

The ID of the request sent to Amazon by this operation.

**returnResult**

The snapshot identifier.

**snapshotId**

The snapshot identifier.

**status**

The status of the snapshot (for example, **pending**, **completed**, or **error**).

**volumeId**

The ID of the volume from which the snapshot was created.

**startTime**

The timestamp when the snapshot was initiated.

**progress**

The progress of the snapshot creation process, in percentage.

**Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

### ***Modify Snapshot Attribute***

The **Modify Snapshot Attribute** operation modifies the (permissions) attribute of a snapshot. You can make your snapshots public or grant permission to them for a given user.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**snapshotId**

The ID of the snapshot you want to modify.

**attribute**

The attribute to modify. Currently, the only attribute you can modify is **createVolumePermission**.

**operationType**

The operation to perform on the attribute. The valid values are **add** and **remove**. You should specify this input only if the **attribute** input is set to **createVolumePermission**.

**userIds**

A list of users you want to grant to or revoke access from. This list can only contain IDs that are numeric. You should only specify this input if the **attribute** input is set to **createVolumePermission**.

**groups**

A list of the groups you want to grant to or revoke access from. According to Amazon, the only currently supported value is **all**. Other values may be supported in the future. You should specify this input only if the **attribute** input is set to **createVolumePermission**.

**delimiter**

The delimiter for the elements of the inputs that are lists.

The operation returns the following:

**requestId**

The ID of the request sent to Amazon by this operation.

**returnResult**

Returns **true** if the operation completed successfully or **false** if it fails.

**Notes:**

- If the **attribute** input is set to **createVolumePermission**, the **operationType** input and at least one of the **userIds** or **groups** inputs should not be empty.
- Like all Amazon EC2 operations, the results might not be visible instantly.

- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

## ***Reset Snapshot Attribute***

The **Reset Snapshot Attribute** operation resets an attribute of a snapshot to its default value.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### **snapshotId**

The ID of the snapshot you want to modify.

### **attribute**

The attribute to reset. The only currently supported value is **createVolumePermission**. Other values may be supported in the future.

The operation returns the following:

### **requestId**

The ID of the request sent to Amazon by this operation.

### **returnResult**

Returns **true** if the operation completed successfully or **false** if it fails.

### **Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

## **Volumes**

### ***Attach Volume***

The **Attach Volume** operation attaches an Amazon EBS volume to a running instance and exposes it as the specified device.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### **volumeId**

The ID of the Amazon EBS volume. The volume must not be attached to any instance.

### **instanceId**

The ID of the instance to which the volume attaches. The volume and instance must be within the same availability zone, and the instance must be running.

### **deviceName**

Specifies how the device is exposed to the instance (for example, **/dev/sdh**, **/dev/sdi**, **xvdh**, or **xvdi**).

The operation returns the following:

### **requestId**

The ID of the request you have sent to Amazon by using this operation.

### **returnResult**

The status of the attaching process (for example, **attaching**).

## **attachTime**

The time the process was launched.

## **Notes:**

- The Windows devices are **xvda** through **xvdp**. **xvda** and **xvdb** are reserved by the operating system and **xvdc** is assigned to drive C:\. For some instance types, devices **xvdd** through **xvde** might also be reserved by the instance stores.
- The Linux and UNIX devices are **/dev/sdf** through **/dev/sdl**. The devices **/dev/sda1**, **/dev/sda2**, **/dev/sdb**, **/dev/sdc**, **/dev/sdd**, and **/dev/sde** are reserved.
- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.
- If the response of the operation is **attaching**, this does not guarantee that the volume attached. Use the **Get Volume Details** operation to check the status of the volume. If the volume has not attached it may be because you specified an invalid device name.

## **Create Volume**

The **Create Volume** operation creates a new Amazon EBS volume.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### **size**

A positive integer that represents the size of the volume, in GBs. This input is required if you are not creating a volume from a snapshot.

### **snapshotId**

The snapshot from which to create the new volume. This input is required if you do not specify the **size** input.

### **availabilityZone**

The availability zone in which to create the new volume.

The operation returns the following:

### **requestId**

The ID of the request you have sent to Amazon by using this operation.

### **returnResult**

The ID of the newly created volume.

### **volumeId**

The ID of the newly created volume.

### **size**

The size of the newly created volume, in GBs.

### **status**

The volume state (for example, **creating**, **available**, **in-use**, **deleting**, or **error**).

### **createTime**

The timestamp when volume creation was initiated.

### **availabilityZone**

The availability zone in which the volume was created.

**snapshotId**

The snapshot from which the volume was created, if applicable.

**Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

### *Delete Volume*

The **Delete Volume** operation deletes an Amazon EBS volume.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**volumeId**

The ID of the Amazon EBS volume. The volume must not be attached to any instance.

The operation returns the following:

**requestId**

The ID of the request you have sent to Amazon by using this operation.

**returnResult**

The status of the process (for example, **deleting**).

**Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

### *Describe Volumes*

The **Describe Volumes** operation describes the status of your Amazon EBS volumes. If you specify a list of volume IDs, only those volumes are described. Otherwise, all volumes that you own are described. This operation does not return information about the attachment state; you must use the **Get Volume Details** operation for that.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**volumeIds**

An optional list of volume IDs. If you specify a list of volume IDs, only those volumes will be described.

**delimiter**

The delimiter for the list of volume IDs. The default is a comma (,).

**rowDelimiter**

The delimiter used between rows in the output table. The default is the line separator of the current operating system.

**colDelimiter**

The delimiter used between columns in the output table. The default is a comma (,).

The operation returns the following:

**requestId**

The ID of the request you have sent to Amazon by using this operation.

## returnResult

A table with the following columns: the volume ID, the volume size, the ID of the snapshot from which the volume was created, the availability zone, the volume's status (for example, **creating**, **available**, **in-use**, **deleting**, or **error**). Each row corresponds to a different volume. If a column is empty, the result is null.

## Notes:

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

## *Detach Volume*

The **Detach Volume** operation detaches an Amazon EBS volume from an instance.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### volumeId

The ID of the Amazon EBS volume to be detached.

### instanceId

The ID of the instance to which the volume is attached.

### deviceName

Specifies how the device is exposed to the instance (for example, **/dev/sdh**, **/dev/sdi**, **xvdh**, or **xvdi**).

### force

Set to **true** if a previous detachment attempt did not occur cleanly. This option can lead to data loss or a corrupted file system. The default value is **false**. If you leave this input blank, it uses the default value.

The operation returns the following:

### requestId

The ID of the request you have sent to Amazon by using this operation.

### returnResult

The status of the process (for example, **detaching**).

### attachTime

The time the association was created.

## Notes:

- The Windows devices are **xvda** through **xvdp**. **xvda** and **xvdb** are reserved by the operating system, and **xvdc** is assigned to drive C:\.
- The Linux and UNIX devices are **/dev/sdf** through **/dev/sdl**. The devices **/dev/sda1**, **/dev/sda2**, **/dev/sdb**, **/dev/sdc**, **/dev/sdd**, and **/dev/sde** are reserved.
- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

## *Get Volume Details*

The **Get Volume Details** operation describes the status of an EBS volume.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**volumeId**

The ID of the volume to describe.

**rowDelimiter**

The delimiter used between rows in the **attachmentSet** output table. The default is the line separator of the current operating system.

**colDelimiter**

The delimiter used between columns in the **attachmentSet** output table. The default is a comma (,).

The operation returns the following:

**requestId**

The ID of the request you have sent to Amazon by using this operation.

**returnResult**

The volume ID.

**volumeId**

The volume ID.

**size**

The volume size.

**snapshotId**

The snapshot from which the volume was created. This field may be empty.

**availabilityZone**

The availability zone in which the volume was created.

**status**

The volume state (for example, **creating**, **available**, **in-use**, **deleting**, or **error**).

**createTime**

The timestamp from when the volume was created. This is converted to the RAS machine timezone.

**attachmentSet**

A table that lists the instances to which the volume is attached. Currently, you can attach a volume to just one instance, but this table is used in case this changes in the future. The table has the following columns: the instance ID, the device through which the volume is exposed to the instance (for example, **/dev/sdf** on a Linux system or **xvdf** on a Windows system), the attachment state, the timestamp when the volume was attached, and the volume ID. Each row corresponds to a different instance.

**Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

## Elastic IPs

### Allocate New Elastic Ip

The **Allocate New Elastic Ip** operation acquires an elastic IP address for use with your Amazon EC2 account.

All of the operation's inputs are described in [Common inputs in the integration](#).

The operation returns the following:

#### **requestId**

The ID of the request you have sent to Amazon by using this operation.

#### **returnResult**

Your new elastic IP address.

#### **Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

### Associate Elastic Ip

The **Associate Elastic Ip** operation associates an elastic IP address with an instance. If the IP address is currently assigned to another instance, it is reassigned to the new instance. The old instance will not have a public DNS name for a few minutes. If you call this operation more than once with the same inputs, Amazon EC2 does not return an error.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

#### **ipAddress**

The elastic IP address to use for your instance.

#### **instanceId**

The ID of the instance.

The operation returns the following:

#### **requestId**

The ID of the request you have sent to Amazon by using this operation.

#### **returnResult**

Returns **true** if the operation completes successfully.

#### **Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

### Disassociate Elastic Ip

The **Disassociate Elastic Ip** operation disassociates the specified elastic IP address from the instance to which it is assigned. If the IP address is not currently assigned to an instance, the operation succeeds but does nothing. The instance to which the address was associated will not have a public DNS name for a few minutes.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**ipAddress**

The elastic IP address you want to disassociate.

The operation returns the following:

**requestId**

The ID of the request you have sent to Amazon by using this operation.

**returnResult**

Returns **true** if the operation completes successfully.

**Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

## Enumerate Elastic Ips

The **Enumerate Elastic Ips** operation lists the elastic IPs addresses assigned to your account.

All of the flow's inputs except the following are described in [Common inputs in the integration](#).

**rowDelimiter**

The row delimiter in the results table. The default value is the line separator for the current operating system.

**colDelimiter**

The column delimiter in the results table. The default value is a comma (,).

The flow returns the following:

**requestId**

The ID of the request you have sent to Amazon by using this operation.

**returnResult**

Your elastic IPs in a table with the following columns: the IP Address and the Instance ID associated to that address.

**Example:**

```
174.129.17.27,i-c0ad46a8
```

```
174.129.17.39,null
```

**Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

## Release Elastic Ip

The **Release Elastic Ip** operation releases an elastic IP address associated with your Amazon EC2 account. If you run this operation on an elastic IP address that you do not own (maybe you have released it already), the operation fails. Releasing an IP address automatically disassociates it from any instance with which it is associated. To disassociate an IP address without releasing it, use the **Disassociate Elastic Ip** operation. The instance will not have a public DNS name for a few minutes.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**ipAddress**

The IP address to release from your account.

The operation returns the following:

**requestId**

The ID of the request you have sent to Amazon by using this operation.

**returnResult**

Returns **true** if the operation completes successfully.

**Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

## Images

### Enumerate Images

The **Enumerate Images** operation returns a list of the Amazon EC2 images that are accessible to you. Images include AMIs (Amazon Machine Images), AKIs (Amazon Kernel Images), and ARIs (Amazon RAM disk Images). You can filter the results to include only images:

- Of a certain platform
- Published by a particular user
- Based on the criterion that they are accessible only to you or that they are public

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**imageOwners**

A list of the IDs of the owners whose images you want to see (for example, **099720109477,3272169288991**). If you leave this input empty, all of the images are returned.

**executableBy**

The accessibility of the images (for example, **all** or **self**).

**colDelimiter**

The column delimiter for the results table.

**rowDelimiter**

The row delimiter for the results table.

**delimiter**

The delimiter for the elements of the results that are lists.

The operation returns the following:

**requestId**

The ID of the request you have sent to Amazon by using this operation.

**returnResult**

The images in a table with the following columns: **AMI ID**, **AMI name**, **Manifest Location**, **Description**, **Visibility**, **Architecture**, **OwnerID**, **Owner Alias Name**, **Image Type**, **Image**

**State, Kernel ID, Ramdisk ID, Platform, State Reason Name, Root Device Name, and Root Device Type.**

**Example:**

```
aki-00806369,null,karmic-kernel-zul/ubuntu-kernel-2.6.31-300-ec2-i386-20091001-  
test-04.manifest.xml,null,true,i386,099720109477,null,available,kernel,null,  
null,null,null,null,instance-store  
aki-00896a69,null,karmic-kernel-zul/ubuntu-kernel-2.6.31-300-ec2-i386-20091002-  
test-04.manifest.xml,null,true,i386,099720109477,null,available,kernel,null,  
null,null,null,null,instance-store  
aki-008b6869,null,redhat-cloud/RHEL-5-Server/5.4/x86_64/kernels/kernel-2.6.18-  
164.x86_64.manifest.xml,null,true,x86_64,432018295444,null,available,kernel,null,  
null,null,null,null,instance-store  
aki-00f41769,null,karmic-kernel-zul/ubuntu-kernel-2.6.31-301-ec2-i386-20091012-  
test-06.manifest.xml,null,true,i386,099720109477,null,available,kernel,null,  
null,null,null,null,instance-store
```

**Notes:**

- Visibility is **true** if the image is public and **false** if it only has implicit and explicit launch permissions. Platform is not always specified.
- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

## Get Image Details

The **Get Image Details** operation returns details about a particular Amazon image. The image can be AMI (Amazon Machine Image), AKI (Amazon Kernel Image), or ARI (Amazon RAM disk Image).

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**imageId**

The ID of the image you want details about. If you specify a nonexistent ID, the operation fails.

**delimiter**

The delimiter for the elements of the result which are lists.

The operation returns the following:

**returnResult**

The image ID of the Amazon image you want details about.

**name**

The name of the Amazon image.

**description**

The description of the Amazon image.

**imageLocation**

The location of the image.

**kernelId**

The operating system kernel associated with the image (only for AMIs, this is optional).

**ramdiskId**

The RAM disk associated with the image (only for AMIs, this is optional).

**platform**

The operating system platform (optional).

**architecture**

The architecture of the image (for example, **i386** or **x86\_64**).

**ownerId**

The account number of the image owner.

**status**

The status of the image. If the operation returns **available**, the image is successfully registered and available for launching. If the operation returns **deregistered**, the image is deregistered and no longer available for launching.

**imageType**

The type of the image (for example, **machine**, **kernel**, or **ramdisk**).

**isPublic**

Returns **true** if this image has public launch permissions and **false** if it only has implicit and explicit launch permissions.

**productCodes**

The product codes of the image.

**rootDeviceType**

The root device type used by the AMI. The AMI can use an Amazon EBS or instance store root device.

**rootDeviceName**

The root device name (for example, **/dev/sda1**).

**stateReason**

The message for the reason for the state change.

**requestId**

The ID of the request you have sent to Amazon by using this operation.

**Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

## Modify Image Attribute

The **Modify Image Attribute** operation modifies an attribute of an AMI (Amazon Machine Image). This allows you to make your AMIs public, grant permission to an AMI for a given user, or associate a product code with an AMI.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**amiId**

The ID of the AMI you want to modify.

**attribute**

The attribute to modify (the valid values are **launchPermission** and **productCode**).

### **operationType**

The operation to perform (the valid values are **add** and **remove**). Specify this input only if the **attribute** input is set to **launchPermission**.

### **userIds**

A list of users to whom you want to grant or revoke access. The list can only contain numeric IDs. Specify this input only if the **attribute** input is set to **launchPermission**.

### **groups**

The groups to which you want to grant or revoke access (according to Amazon, the only currently supported value is **all**; other values may be supported in the future). Specify this input only if the **attribute** input is set to **launchPermission**.

### **productCode**

The product code to associate with your AMI. Specify this input only if the **attribute** input is set to **productCode**. According to Amazon, each AMI can have a single product code associated with it. The user must be the owner of the AMI to associate a product code with it.

### **delimiter**

The delimiter for the elements of the inputs that are lists.

The operation returns the following:

### **requestId**

The ID of the request you have sent to Amazon by using this operation.

### **returnResult**

Returns **true** if the operation completes successfully and **false** if it does not.

### **Notes:**

- If the **attribute** input is set to **productCode**, the **productCode** input cannot be empty.
- If the **attribute** input is set to **launchPermission**, the **operationType** input and at least one of the **userIds** or **groups** inputs should not be empty.
- An Amazon DevPay account is needed in order to be able to change the **productCode**.
- **Important:** Once set, the **productCode** cannot be changed or removed.
- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

## **Register Image**

The **Register Image** operation registers an AMI with Amazon EC2. Images must be registered before they can be launched. You can register an image more than once, and have multiple AMI IDs for the same image at the same time.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### **imageLocation**

The full path to your AMI manifest in Amazon S3 storage (for example, **/mybucket/myimage.manifest.xml**).

The operation returns the following:

### **requestId**

The ID of the request you have sent to Amazon by using this operation.

## returnResult

The unique ID of the newly registered machine image (an AMI ID).

### Notes:

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

## Remove Image

The **Remove Image** operation deregisters the specified Amazon Machine Image (AMI). Once deregistered, the AMI cannot be used to launch new instances. The image is not deleted from Amazon S3.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### amiId

The ID of the AMI you want to deregister.

The flow returns the following:

### requestId

The ID of the request you have sent to Amazon by using this operation.

### returnResult

Returns **true** if the operation completes successfully.

### Notes:

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

## Reset Image Attribute

The **Reset Image Attribute** operation resets an attribute of an Amazon Machine Image (AMI) to its default value.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### amiId

The ID of the AMI you want to modify.

### attribute

The attribute to reset (the valid values are **launchPermission** and **productCodes**).

The operation returns the following:

### requestId

The ID of the request you have sent to Amazon by using this operation.

### returnResult

Returns **true** if the operation completes successfully and **false** if it does not.

### Notes:

- The **productCodes** attribute cannot be reset.
- Like all Amazon EC2 operations, the results might not be visible instantly.

- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

## Instances

### Describe Instances

The **Describe Instances** operation returns a list of your Amazon EC2 instances.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

#### **instanceIds**

A list of the instance IDs to describe. If you leave this input empty, all instances are described. The instances are returned in the order that the IDs are given in this input. If you leave this input empty, the instance IDs are returned in the order the instances are stored by Amazon EC2; this is the same order in which the inputs can be seen in the user interface provided by Amazon.

#### **delimiter**

The delimiter used for the **instanceIds** list and for the list of security groups in the response string. The default is a comma (,).

#### **rowDelimiter**

The delimiter that separates two instance descriptions in the output string. The default is the line separator for the current operating system.

#### **colDelimiter**

The delimiter that separates the fields that describe an instance in the result string. The default is a semicolon (;).

The operation returns the following:

#### **requestId**

The ID of the request you have sent to Amazon by using this operation.

#### **returnResult**

Your EC2 instances in a table with the following columns: **Instance Id**, **AMI ID**, **Machine type**, **Status**, **Public DNS**, **Key pair name**, **Ramdisk ID**, **Platform**, **Kernel ID**, **Monitoring**, **Root Device Name**, **Root Device Type**, **State Transition Reason**, **Availability Zone**, and **Security groups**. Column fields are separated by the **colDelimiter** input. If you have no instances, an empty string is returned.

#### **Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

### Get Instance Details

The **Get Instance Details** operation returns details about a specific Amazon EC2 instance.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

#### **instanceIdentifier**

The ID of the instance you want details about. If you specify a nonexistent ID, the operation fails.

**delimiter**

The delimiter used for the **instanceIds** list and for the list of security groups in the response string. The default is a comma (,).

The operation returns the following:

**returnResult**

The **instanceId** of the specified instance.

**imageId**

The ID of the AMI used for this instance.

**kernelId**

The operating system kernel associated with the instance (optional).

**platform**

The operating system platform (optional).

**publicDns**

The public hostname of the instance (contactable from outside the Amazon EC2 network).

**privateDns**

The private hostname of the instance (can only be used inside the Amazon EC2 network).

**launchTime**

The time the instance was launched.

**status**

The status of the instance (for example, **running**, **pending**, or **terminated**).

**transitionReason**

The reason the instance entered this state.

**availabilityZone**

The availability zone in which the instance is located.

**machineType**

The type of the instance (for example, **m1.small** or **c1.xlarge**).

**ownerId**

The account number of the AMI owner.

**ramdiskId**

The RAM disk associated with the image (optional).

**keyPairName**

The key pair name if this instance was launched with an associated key pair.

**launchIndex**

The order in which the instance was launched when launching more than one instance in one reservation.

**monitoring**

The monitoring status.

**reservationId**

The reservation ID used to launch the instance.

**productCodes**

The list of product codes.

**groupSet**

A list of the security groups to which the instance belongs, separated by the **delimiter** input.

**requestId**

The ID of the request sent to Amazon.

**Notes:**

- Instances that are terminated or starting up may not have a public DNS name or other attributes.
- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

## Get System Log

The **Get System Log** operation retrieves the console output (the system log) of a running instance. Instance console output is buffered and posted shortly after instance boot, reboot, and termination. Amazon EC2 preserves the most recent 64 KB of output which is available for at least one hour after the most recent post.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**instanceId**

The ID of the instance you want to query.

The operation returns the following:

**requestId**

The ID of the request you have sent to Amazon by using this operation.

**returnResult**

The console output.

**updateTime**

The time the output was last updated.

**Notes:**

- This operation may also work on instances which were terminated a few days before. Use the **Get Instance Details** operation to verify that the queried instance is still alive.
- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

## Monitor Instance

The **Monitor Instance** operation enables Amazon CloudWatch monitoring for a running instance.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**instanceIdentifier**

The ID of the instance you want to monitor.

The operation returns the following:

**requestId**

The ID of the request you have sent to Amazon by using this operation.

**returnResult**

The state of the monitoring after the execution of this operation (for example, **pending**).

**Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

## Restart Instance

The **Restart Instance** operation requests a reboot of one instance. The operation succeeds if the instance is valid and belongs to you. Requests to reboot terminated instances are ignored.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**instanceIdentifier**

The ID of the instance you want to reboot.

The operation returns the following:

**requestId**

The ID of the request you have sent to Amazon by using this operation.

**returnResult**

Returns **true** if the instance is being restarted. This may take some time and the instance's status may remain **running** while it is being restarted.

**Notes:**

- We recommend that you verify (for instance, using the **Get Instance Details** operation) that the instance you want to restart exists and that it is in the **running** state. Otherwise, the operation's **returnResult** may be **true** but nothing may have happened.
- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

## Run Instances

The **Run Instances** operation launches a specified number of instances of an AMI for which you have permissions. If Amazon EC2 cannot launch the minimum number AMIs you request, no instances are launched. If there is insufficient capacity to launch the maximum number of AMIs you request, Amazon EC2 launches the minimum number specified for each AMI and allocates the remaining available instances using round robin.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**amiId**

The unique ID of the Amazon Machine Image you want to launch.

**minCount**

The minimum number of instances you want to launch from the specified AMI.

**maxCount**

The maximum number of instances you want to launch from the specified AMI.

**kernelId**

The ID of the kernel with which to launch the instance.

**virtualName**

The virtual device name (for example, **ami**, **root**, **swap**, or **ephemeralN**).

**deviceName**

The device name (for example, **/dev/sdh**).

**ramdiskId**

The ID of the RAM disk with which to launch the instance, required by some kernels.

**keyName**

The name of one of your existent key pairs, used to access your instance. Launching public images without a key pair ID leaves them inaccessible.

**securityGroups**

The name of the security group or groups of which your instances will be part. If you do not specify a security group, the **default** group is used.

**instanceType**

The instance type. The valid values are **m1.small**, **m1.large**, **m1.xlarge**, **c1.medium**, and **c1.xlarge**. If the **amiId** input specifies a 64 bit image, then this input should be provided with one of the following values: **m1.large**, **m1.xlarge**, or **c1.xlarge**. If the **amiId** input specifies a 32 bit image, then this input can be left empty or given one of the following values: **m1.small** or **c1.medium**. In this case, if the input is left empty, a small instance is created by default.

**placementZone**

The placement constraints (availability zones) for launching the instances. By default, Amazon selects an availability zone automatically.

**monitoring**

Specifies whether the instance should be monitored after its launch. The valid values are **true** or **enabled** if the instance should be monitored after launch and **false** or **disabled** if not. The default is **false**. If you do not specify a value, the instance is not monitored after starting.

**delimiter**

The delimiter for returned lists of IDs and the **securityGroups** input. The default is a comma **,**.

**snapshotId**

The ID of the snapshot.

**volumeSize**

The size of the volume in GBs. This input is required if you are not creating a volume from a snapshot.

**deleteOnTermination**

Specifies whether the Amazon EBS volume is deleted on instance termination. The valid values are **true** and **false**.

**subnetId**

The Amazon VPC subnet ID within which to launch the instance(s) for Amazon Virtual Private Cloud.

### **disableApiTermination**

Specifies whether the instance can be terminated using the APIs. You must modify this attribute before you can terminate any "locked" instances from the APIs. The valid values are **true** and **false**. A value of **false** means the instance is unlocked.

### **instanceInitiatedShutdownBehavior**

Specifies whether the instance's Amazon EBS volumes are stopped or terminated when a shutdown is issued from with the instance. The valid values are **true** and **false**.

The operation returns the following:

#### **returnResult**

A list containing the IDs of your newly started instances.

#### **reservationID**

The reservation ID used to launch the instances.

#### **requestId**

The ID of the request you have sent to Amazon by using this operation.

#### **requesterId**

The ID of the requester who sent this request to Amazon.

#### **Notes:**

- The block device mappings exposed to the instance are made up of a virtual name (**virtualName**) and a device name (**deviceName**). Virtual names include:
  - **ami**—The root file system device, as seen by the instance.
  - **root**—The root file system device, as seen by the kernel.
  - **swap**—The swap device, as seen by the instance.
  - **ephemeralN**—The Nth ephemeral store.Virtual names are used together with device names. For example: **ami=sda1, root=/dev/sda1, ephemeral0=sda2, or swap=sda3**.
- If you do not specify a value for the **minCount** and **maxCount** inputs, one instance is launched.
- The architecture of your AMI, RAM disk, and kernel must match. Even then, there is no guarantee that a given combination will function correctly or boot successfully if they weren't designed to be used together.
- Instances usually enter the **pending** state after this operation. They do not have DNS names and other attributes for at least a few minutes.
- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." message is returned.
- If any of the AMIs has a product code attached for which the user has not subscribed, the operation fails.

## **Run Instances From Template**

The **Run Instances From Template** flow launches a specified number of instances with the same configuration as a given one. If Amazon EC2 cannot launch the minimum number of instances you request, no instances are launched. If there is insufficient capacity to launch the maximum number of instances you request, Amazon EC2 launches the minimum number specified and allocates the remaining available instances using round robin.

All of the flow's inputs except the following are described in [Common inputs in the integration](#).

**instanceIdentifier**

The ID of the template instance. The following configuration characteristics of the instance are used when launching the new instances: AMI ID, availability zone, machine type, kernel ID, ramdisk ID, monitoring, key pair name, and security groups.

**minCount**

The minimum number of instances you want to launch.

**maxCount**

The maximum number of instances you want to launch.

The flow returns the following:

**returnResult**

A list containing the IDs of your newly started instances.

**Notes:**

- If you do not specify a value for the **minCount** and **maxCount** inputs, one instance is launched.
- Instances usually enter the **pending** state after this flow. They do not have DNS names and other attributes for at least a few minutes.
- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.
- If any of the AMIs has a product code attached for which the user has not subscribed, the operation fails.

## Start Instance

The **Start Instance** operation starts an instance that uses an Amazon EBS volume as its root device. When an instance is stopped, the compute resources are released and you are not billed for hourly instance usage. However, the root partition Amazon EBS volume remains, continuing to persist the data, and you are charged for Amazon EBS volume usage. You can restart the instance at any time.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**instanceId**

The instance ID.

The operation returns the following:

**returnResult**

The result of the request.

**requestId**

The ID of the request you have sent to Amazon by using this operation.

**name**

The state of the instance.

**Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.

- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

## Stop Instance

The **Stop Instance** operation stops an instance that uses an Amazon EBS volume as its root device. Instances that use Amazon EBS volumes as their root devices can be quickly stopped and started. When an instance is stopped, the compute resources are released and you are not billed for hourly instance usage. However, your root partition Amazon EBS volume remains, continuing to persist your data, and you are charged for Amazon EBS volume usage. You can restart your instance at any time.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### **instanceIdentifier**

The ID of the instance you want to terminate.

The operation returns the following:

### **requestId**

The ID of the request you have sent to Amazon by using this operation.

### **returnResult**

The state of the instance after the operation (for example, **shutting-down**).

### **previousState**

The state the instance was in before you stopped it (for example, **running**).

### **Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." is returned.

## Terminate Instance

The **Terminate Instance** operation shuts down an instance. If you terminate an instance more than once, each call succeeds. Terminated instances remain visible (for instance, when using the **Describe Instances** operation) after termination for approximately one hour.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### **instanceIdentifier**

The ID of the instance you want to terminate.

The operation returns the following:

### **requestId**

The ID of the request you have sent to Amazon by using this operation.

### **returnResult**

The state of the instance after the operation (for example, **shutting-down**).

### **previousState**

The state the instance was in before you stopped it (for example, **running**).

### **Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.

- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." message is returned.

## Unmonitor Instance

The **Unmonitor Instance** operation disables Amazon CloudWatch monitoring for a running instance.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### **instanceIdentifier**

The ID of the instance for which you want to disable monitoring.

The operation returns the following:

### **requestId**

The ID of the request you have sent to Amazon by using this operation.

### **returnResult**

The state of the monitoring after the execution of this operation (for example, **disabling**).

### **Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." message is returned.

## Key Pairs

### Allocate New Key Pair

The **Allocate New Key Pair** operation creates a new 2048-bit RSA key pair with the specified name. The private key is returned as an unencrypted PEM encoded PKCS#8 private key. If a key with the specified name already exists, Amazon EC2 returns an error.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### **keyName**

The name of the new key pair.

The operation returns the following:

### **requestId**

The ID of the request you have sent to Amazon by using this operation.

### **returnResult**

The key material of your RSA private key.

### **keyFingerprint**

An SHA-1 digest of the DER encoded private key.

### **Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." message is returned.

## Delete Key Pair

The **Delete Key Pair** operation deletes the specified key pair by removing the public key from Amazon EC2. You must own the key pair. The operation succeeds if you call it more than once on a given key and even if the key does not exist. We recommend that you check that the key you want to delete exists by using the **Enumerate Key Pairs** operation.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### keyName

The name of the key pair you want to delete.

The operation returns the following:

### requestId

The ID of the request you have sent to Amazon by using this operation.

### returnResult

Returns **true** if the operation succeeds.

### Notes:

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." message is returned.

## Enumerate Key Pairs

The **Enumerate Key Pairs** operation lists the key pairs that are available to you.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### rowDelimiter

The row delimiter in the results table. The default value is the line separator for the current operating system.

### colDelimiter

The column delimiter in the results table. The default value is a comma (,).

The operation returns the following:

### requestId

The ID of the request you have sent to Amazon by using this operation.

### returnResult

Your key pairs in a table with the following columns: **Key name** and **Fingerprint**.

#### Example:

```
testkey2,8c:0a:1a:c8:a5:07:cb:3d:0e:60:3f:00:7d:a1:e9:c4:4f:28:e4:7b  
minFedora,8b:79:f0:6d:6d:4c:73:80:c0:1a:9e:e3:81:f3:cc:3f:33:20:cf:89
```

### Notes:

- In case no key pair exists, the operation fails with the message "No items found!".
- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." message is returned.

## Security groups

### Allow Access To Security Group

The **Allow Access To Security Group** operation adds new permissions to the given security group. You can specify the new permissions by providing the **sourceSecurityGroupName** and the **sourceSecurityGroupOwnerId** inputs, or the source **cidrIp**, **ipProtocol**, **fromPort**, and **toPort** inputs.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

#### **groupName**

The name of the group to modify.

#### **sourceSecurityGroupName**

The name of the source security group.

#### **sourceSecurityGroupOwnerId**

The user ID of the owner of the source security group.

#### **ipProtocol**

The IP protocol used. The valid values are **tcp**, **udp**, and **icmp**.

#### **fromPort**

The start of the port range for the TCP and UDP protocols, and the type number for ICMP. For ICMP, **-1** indicates a wildcard (any ICMP type number).

#### **toPort**

The end of the port range for the TCP and UDP protocols, and the type number for ICMP. For ICMP, **-1** indicates a wildcard (any ICMP type number).

#### **cidrIp**

The source IP in CIDR format.

The operation returns the following:

#### **requestId**

The ID of the request you have sent to Amazon by using this operation.

#### **returnResult**

Returns **true** when the operation is successful and **false** when it is not.

#### **Notes:**

- If you add a combination of access parameters that already exists, the operation fails.
- If new permissions are specified by a source group name, any machine from the **sourceSecurityGroupName** input can connect to any machine from the **groupName** input, using any of the TCP, UDP, or ICMP protocols.
- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." message is returned.

### Create Security Group

The **Create Security Group** operation creates a new security group.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**groupName**

The name of the new group. This name must be unique per account.

**groupDescription**

The description of the group. This is only informational, and can contain alphanumeric characters, spaces, dashes, and underscores.

The operation returns the following:

**requestId**

The ID of the request you have sent to Amazon by using this operation.

**returnResult**

Returns **true** when the operation is successful or **false** if it is not.

**Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." message is returned.

## Delete Security Group

The **Delete Security Group** operation deletes a security group.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**groupName**

The name of the security group you want to delete.

The operation returns the following:

**requestId**

The ID of the request you have sent to Amazon by using this operation.

**returnResult**

Returns **true** when the operation is successful or **false** if it is not.

**Notes:**

- If you attempt to delete a security group which used to exist but has been recently deleted, the operation might go to **Success**. This may happen because Amazon remembers a deleted security group for a certain amount of time.
- If you attempt to delete a security group that contains instances, a fault is returned.
- If you attempt to delete a security group that is referenced by another security group, a fault is returned. For example, if Security Group B has a rule that allows access from Security Group A, Security Group A cannot be deleted until you remove the rule.
- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." message is returned.

## Describe Security Groups

The **Describe Security Groups** operation describes the security groups that you own. By default, this operation returns information about all of your security groups, but you can specify a list of group names to restrict the results to only those specified.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**groupNames**

The names of the groups that you want to describe. If you leave this input blank, all of your owned groups are described.

**delimiter**

The delimiter for the list of group names. The default is a comma (,).

**rowDelimiter**

The delimiter used between rows in the output table. The default is the line separator of the current operating system.

**colDelimiter**

The delimiter used between columns in the output table. The default is a comma (,).

The operation returns the following:

**requestId**

The ID of the request you have sent to Amazon by using this operation.

**returnResult**

A table listing the security groups, one in each row. The table has the following columns: **group name**, **group owner id**, and **group description**.

**groupNames**

A list of the group names of your security groups.

**Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." message is returned.

## Get Security Group Details

The **Get Security Group Details** operation returns information about a specific security group that you own.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**groupName**

The name of the security group you want to describe.

**rowDelimiter**

The delimiter used between rows in the output table. The default is the line separator of the current operating system.

**colDelimiter**

The delimiter used between columns in the output table. The default is a comma (,).

The operation returns the following:

**requestId**

The ID of the request you have sent to Amazon by using this operation.

**returnResult**

The name of the group.

**groupName**

The name of the group.

**ownerId**

The ID of the owner of the group.

**groupDescription**

The description of the group.

**ipPermissions**

A table describing the permissions for this group, one in each row. The table has the following columns:

- **ip protocol**
- **from port** The start of the port range for the TCP and UDP protocols, or an ICMP type number (an ICMP type number of **-1** indicates a wildcard—any ICMP type number).
- **to port** The end of the port range for the TCP and UDP protocols, or an ICMP code (**-1** indicates a wildcard).
- **source security group** The name of the source security group.
- **user id** The user ID of the owner of the source security group.
- **source IP in CIDR notation**

**Notes:**

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." message is returned.

## Revoke Access To Security Group

The **Revoke Access To Security Group** operation removes a permission from the specified security group. You can specify the permission by using the **sourceSecurityGroupName** or the **sourceSecurityGroupOwnerId** input, or by using the **cidrIp**, **ipProtocol**, **fromPort**, and **toPort** inputs.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**groupName**

The name of the security group.

**sourceSecurityGroupName**

The name of the source security group.

**sourceSecurityGroupOwnerId**

The user ID of the owner of the source security group.

**ipProtocol**

The IP protocol used. The valid values are **tcp**, **udp**, and **icmp**.

**fromPort**

The start of the port range for the TCP and UDP protocols, and the type number for ICMP. For ICMP, **-1** indicates a wildcard (any ICMP type number).

**toPort**

The end of the port range for the TCP and UDP protocols, and the type number for ICM. For ICMP, **-1** indicates a wildcard (any ICMP type number).

**cidrIp**

The source IP in CIDR format.

The operation returns the following:

**requestId**

The ID of the request you have sent to Amazon by using this operation.

**returnResult**

Returns **true** when the operation is successful or **false** if it is not.

**Notes:**

- This operation goes to success even in situations when no permission is revoked or the permission does not exist. The only situation when a permission is actually revoked is when the inputs match exactly the inputs used to create the permission (for example, the inputs given to the **Allow Access To Security Group** operation).
- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." message is returned.

## Windows

### Bundle Windows Instance

The **Bundle Windows Instance** operation bundles the Windows instance. The bundling process creates a new image from a running instance and stores the AMI data in S3. Once it is bundled, you must register the image using the **Register Image** operation prior to launching it.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**instanceId**

The ID of the instance you want to bundle.

**storageBucket**

The Amazon S3 bucket in which to store the AMI. You must specify a bucket that you already own. If you specify a bucket that belongs to someone else, Amazon EC2 returns an error.

**storagePrefix**

A prefix that specifies the beginning of the file name of the AMI. Your image is stored in the bucket specified by the **storageBucket** input under the name `<storagePrefix>.manifest.xml`.

**s3AccessKeyId**

The ID of the secret access key associated with your S3 AWS account, if it is different from the EC2 account. If you do not specify the **s3AccessKeyId** input, the operation uses the EC2 access key ID.

**s3AccessKey**

The secret access key associated with your S3 AWS account, if it is different from the EC2 account. If you do not specify the **s3AccessKey** input, the operation uses the EC2 access key.

The operation returns the following:

**requestId**

The ID of the request you have sent to Amazon by using this operation.

**returnResult**

The Bundle ID by that identifies the bundle task.

**status**

The state of the bundle task (for example, **pending**).

**startTime**

The time the bundle task started.

**Notes:**

- You cannot start another bundle task on an instance until the running task is terminated. Also, if the path you specified for the image already exists, any images under that name may be overwritten.
- The **s3AccessKeyId** and **s3AccessKey** inputs work together. Therefore, you must specify values for both of them or leave them both.
- The operation succeeds for an invalid **s3AccessKey** creating a bundle task, but during the bundling process, the task fails with the following error message: "SignatureDoesNotMatch(403)- The request signature we calculated does not match the signature you provided. Check your key and signing method."
- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." message is returned.

## Cancel Bundle Task

The **Cancel Bundle Task** operation aborts a bundle task.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**bundleId**

The ID of the bundle task. If you specify a nonexistent ID, the operation fails. If you specify the ID of an already terminated bundle task, the operation may succeed but nothing will happen.

The operation returns the following:

**returnResult**

The instance ID of the instance that was being bundled.

**startTime**

The time the task was launched.

**updateTime**

The time the status of the task was last updated.

**status**

The status of the task (for example, **canceling** or **failed**).

**progress**

The progress of the task (**0%** - **100%**).

**bucket**

The S3 bucket where the image would have been placed.

**prefix**

The name under which the image would have been placed in the bucket.

## requestId

The ID of the request you have sent to Amazon by using this operation.

### Notes:

- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." message is returned.

## Enumerate Bundle Tasks

The **Enumerate Bundle Tasks** operation returns a list of your bundle tasks in a table.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### rowDelimiter

The row delimiter in the results table. The default value is the line separator for the current operating system.

### colDelimiter

The column delimiter in the results table. The default value is a comma (,).

The operation returns the following:

### requestId

The ID of the request you have sent to Amazon by using this operation.

### returnResult

Your bundle tasks in a table having the following columns: **Bundle ID**, **Instance ID**, **Status**, **Start time**, and **Progress**. The column fields are separated by **colDelimiter** and the rows by **rowDelimiter**.

#### Example:

```
bun-49789d20,i-d63cd5be    complete,2009-08-25T10:04:55.000Z,null
bun-4f789d26,i-d63cd5be    failed,2009-08-25T10:58:57.000Z,null
```

### Notes:

- Terminated bundle tasks appear in this list for an undetermined interval of time after they are terminated.
- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." message is returned.

## Get Bundle Task Details

The **Get Bundle Task Details** operation returns details about a specified bundle task.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### bundleId

The ID of the bundle task. If you specify a nonexistent ID, the operation fails.

The operation returns the following:

### returnResult

The instance ID of the instance that is being bundled.

**startTime**

The time the task was launched.

**updateTime**

The time the status of the task was last updated.

**status**

The status of the task (for example, **running**, **complete**, or **failed**).

**progress**

The progress of the task (**0%** - **100%**).

**bucket**

The S3 bucket where the image will be placed.

**prefix**

The name under which the image will be placed in the bucket.

**requestId**

The ID of the request you have sent to Amazon by using this operation.

**Notes:**

- Terminated bundle tasks can be queried for an undetermined interval of time after they are terminated.
- Like all Amazon EC2 operations, the results might not be visible instantly.
- Since all Amazon requests must include a timestamp, the RAS machine system clock should be properly set, otherwise a "Request has expired. Timestamp date is ..." message is returned.

## EC2 Dashboard

The **EC2 Dashboard** flow displays the EC2 dashboard information: the number of instances running or stopped, elastic IPs, EBS volumes, EBS snapshots, key pairs, and security groups.

All of the flow's inputs except the following are described in [Common inputs in the integration](#).

**notifyMethod**

The method of notification. The valid values are **Write to File**, **Email**, **Display**, and **None**.

The flow returns the following:

**returnResult**

The dashboard information.

## Troubleshooting

This section provides troubleshooting procedures and tools you can use to solve problems you may encounter while using this integration. It also includes a list of the error messages you may receive while using the integration and offers descriptions and possible fixes for the errors.

### General troubleshooting procedures and tools

When troubleshooting issues related to the EC2 integration, there are several steps that you should verify:

1. Verify that the same operation works when it is performed through the EC2 console using the same parameters.
2. Verify that your access key and proxy parameters are correct.
3. Verify that the RAS machine system clock is properly set since all Amazon queries must include a timestamp.

## **Error messages**

This section lists the error messages you may receive while using this integration.

**Attribute productCodes can not be reset currently.**

**AWS was not able to validate the provided access credentials.**

**Either both proxy port and proxy host must be null or none of them.**

**Failed to open HTTP connection.**

**fromPort was invalid. Must be numerical and between 0 and 65535.**

**If attribute=productCodes, productCode must be set!**

**Invalid attribute item value '###' for UserGroup item type. Error code: InvalidAMIAttributeItemValue.**

**Invalid attribute item value '###' for userId item type. Error code: InvalidAMIAttributeItemValue.**

**Invalid attribute.**

**Invalid bundle ids: ###. Error code: InvalidBundleID.NotFound.**

**Invalid id: '###' (expecting "ami-..."). Error code: InvalidAMIID.Malformed.**

**Invalid id: '###' (expecting "i-..."). Error code: InvalidInstanceID.Malformed.**

**Invalid manifest path: '###'. Error code: InvalidManifest.**

**Invalid operationType.**

**Invalid user id: '###'. Error code: InvalidUserID.Malformed.**

**Must supply either only user/group pair permission inputs (SourceSecurityGroupName and SourceSecurityGroupOwnerId) or only CIDR IP permission inputs (IpProtocol, FromPort, ToPort, CidrIp).**

**Not authorized for image: ###. Error code: AuthFailure.**

**Possible values for input "force" are true or false.**

**The address '###' does not belong to you. Error code: AuthFailure.**

**The AMI ID '###' does not exist. Error code: InvalidAMIID.NotFound.**

**The following input must be a valid integer: size.**

**The given CIDR IP address is invalid. It must have the following format: [0-255].[0-255].[0-255].[0-255]/[0-32].**

The given IP address is invalid. It must have the following format: [0-255].[0-255].[0-255].[0-255].

The given IpProtocol is invalid. Must be one of 'tcp', 'udp' or 'icmp'.

The instance ID '###' does not exist. Error code: InvalidInstanceId.NotFound.

The instance IDs '###' do not exist. Error code: InvalidInstanceId.NotFound.

The keypair '###' already exists. Error code: InvalidKeyPair.Duplicate.

The permission '###' has already been authorized on the specified group. Error code: InvalidPermission.Duplicate.

The request must contain the parameter KeyName. Error code: MissingParameter.

The request signature we calculated does not match the signature you provided. Check your AWS Secret Access Key and signing method. Consult the service documentation for details.

The security group '###' already exists. Error code: InvalidGroup.Duplicate.

The security group '###' does not exist. Error code: InvalidGroup.NotFound.

The security group '###' is reserved. Error code: InvalidGroup.Reserved.

The snapshot '###' does not exist. Error code: InvalidSnapshot.NotFound.

The volume '###' does not exist. Error code: InvalidVolume.NotFound.

The volume size must be an integer between 1 and 1024 inclusive.

The zone '###' does not exist. Error code: InvalidZone.NotFound.

toPort was invalid. Must be numerical and between 0 and 65535.

Value (###) for parameter instance is invalid. Expected: 'i-...'. Error code: InvalidParameterValue.

Value (###) for parameter snapshotId is invalid. Expected: 'snap-...'. Error code: InvalidParameterValue.

Value (###) for parameter volumeId is invalid. Expected: 'vol-...'. Error code: InvalidParameterValue.

## Security

This section describes how security is handled by the Amazon EC2 integration.

The EC2 integration uses the EC2 Query API. To perform its task, each EC2 operation sends a query to Amazon. This query is sent over HTTPS and the Symphony Client library is used for sending the actual requests. Every EC2 operation has 2 inputs: **accessKey** and **accessKeyId**. These are provided by Amazon for every Amazon account.

According to Amazon, in addition to the name of the action and the list of parameters, you must include a signature in every Query request. The signature is created by using the **accessKey** provided by the user. The steps for creating a signature are described on the Amazon Web site at <http://docs.amazonwebservices.com/AWSEC2/2009-04-04/DeveloperGuide/index.html?using-query-api.html>, and are implemented by the operations.

The EC2 integration uses version 2 of the signature. For calculating an RFC 2104-compliant HMAC with the query string created by using the input parameters, the integration use the secret Access Key as the key, and SHA1 as the hash algorithm. You should not perform any special configuration, just provide the **accessKey** and **accessKeyId**.

## Tools

Following are OO tools that you can use with the Amazon EC2 integration:

- **RSFlowInvoke.exe** and **JRSFlowInvoke.jar**

RSFlowInvoke (RSFlowInvoke.exe or the Java version, JRSFlowInvoke.jar) is a command-line utility that allows you to start a flow without using Central (although the Central service must be running). RSFlowInvoke is useful when you want to start a flow from an external system, such as a monitoring application that can use a command line to start a flow.

- **Web Services Wizard (wswizard.exe)**

When you run the Web Services Wizard, you provide it with the WSDL for a given Web service. The WSDL string you provide as a pointer can be a file's location and name or a URL. The Web Services Wizard displays a list of the methods in the API of the Web service that you specify. When you run the wizard, pick the methods you want to use, and with one click for each method you have selected, the wizard creates an HP OO operation that can execute the method. This allows you to use the Web Services Wizard to create operations from your monitoring tool's API.

These tools are available in the %OO\_home%/Studio/tools/ folder.