
AWS Application Discovery Service

User Guide



AWS Application Discovery Service: User Guide

Copyright © 2018 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

What Is AWS Application Discovery Service?	1
Setting Up	3
Step 1: Sign Up for AWS	3
Step 2: Create an IAM User	3
Required IAM User Policies	4
Getting Started	8
Assumptions	8
Accessing AWS Application Discovery Service	8
Start Collecting Data	8
Discovery Connector	9
Data Collected by Discovery Connector	9
Download the Discovery Connector	12
Deploy the Discovery Connector	12
Configure the Discovery Connector	13
Start Data Collection	15
Discovery Agent	16
Data Collected by the Discovery Agent	17
Prerequisites for Agent Installation	19
Agent Installation on Linux	20
Agent Installation on Windows	23
Start Data Collection	26
View & Export Data	28
View Collected Data	28
Export Collected Data	28
Explore Collected Data	30
Console Walkthroughs	32
Main Dashboard	32
Main Dashboard	32
Navigating from the Dashboard and the Navigation Pane	33
Data Collection Tools	34
Starting and Stopping Data Collectors	34
Viewing and Sorting Data Collectors	35
View & Export Server Data	36
Viewing and Sorting Servers	36
Tagging Servers	36
Exporting Server Data	37
Applications	38
Limits	39
Document History	40
AWS Glossary	41

What Is AWS Application Discovery Service?

AWS Application Discovery Service helps you plan your migration to the AWS cloud by collecting usage and configuration data about your on-premises servers. Application Discovery Service is integrated with AWS Migration Hub, which simplifies your migration tracking. After performing discovery, you can view the discovered servers, group them into applications, and then track the migration status of each application from the Migration Hub console. The discovered data can be exported for analysis in Microsoft Excel or AWS analysis tools such as Amazon Athena and Amazon QuickSight.

Using Application Discovery Service APIs, you can export the system performance and utilization data for your discovered servers. You can input this data into your cost model to compute the cost of running those servers in AWS. Additionally, you can export the network connections and process data to understand the network connections that exist between servers. This will help you determine the network dependencies between servers and group them into applications for migration planning.

Application Discovery Service offers two ways of performing discovery and collecting data about your on-premises servers:

- **Agentless discovery** can be performed by deploying the AWS Agentless Discovery Connector (OVA file) through your VMware vCenter. After the Discovery Connector is configured, it identifies virtual machines (VMs) and hosts associated with vCenter. The Discovery Connector collects the following static configuration data: Server hostnames, IP addresses, MAC addresses, disk resource allocations. Additionally, it collects the utilization data for each VM and computes average and peak utilization for metrics such as CPU, RAM, and Disk I/O. You can export a summary of the system performance information for all the VMs associated with a given VM host and perform a cost analysis of running them in AWS.
- **Agent-based discovery** can be performed by deploying the AWS Application Discovery Agent on each of your VMs and physical servers. The agent installer is available for both Windows and Linux operating systems. It collects static configuration data, detailed time-series system-performance information, inbound and outbound network connections, and processes that are running. You can export this data to perform a detailed cost analysis and to identify network connections between servers for grouping servers as applications.

How to decide which discovery tool to use

If you have virtual machines (VMs) that are running in the VMware vCenter environment, you can use the Discovery Connector to collect system information without having to install an agent on each VM. Instead, you load this on-premises appliance into vCenter and allow it to discover all of its hosts and VMs.

The Discovery Connector captures system performance information and resource utilization for each VM running in the vCenter, regardless of what operating system is in use. However, it cannot “look inside” each of the VMs, and as such, cannot figure out what processes are running on each VM nor what network connections exist. Therefore, if you need this level of detail and want to take a closer look at some of your existing VMs in order to assist in planning your migration, you can install the Discovery Agent on an as-needed basis.

Also, for VMs hosted on VMware, you can use both the Discovery Connector and Discovery Agent to perform discovery simultaneously. For details regarding the exact types of data each discovery tool will collect, see [Data Collected by the Discovery Connector \(p. 9\)](#) and [Data Collected by the Discovery Agent \(p. 10\)](#).

[Agent \(p. 17\)](#). A quick view comparison table of the Discovery Connector the Discovery Agent is provided below.

	Discovery Connector	Discovery Agent
Supported server types		
VMware virtual machine	yes	yes
Physical server	no	yes
Deployment		
Per server	no	yes
Per vCenter	yes	no
Collected data		
Static configuration data	yes	yes
VM utilization metrics	yes	no
Time series performance information	no	yes (Export only)
Network inbound/outbound connections	no	yes (Export only)
Running processes	no	yes (Export only)
Supported OS	Any OS running in VMware vCenter (V5.5, V6, & V6.5)	Linux Amazon Linux 2012.03, 2015.03 Ubuntu 12.04, 14.04, 16.04 Red Hat Enterprise Linux 5.11, 6.9, 7.3 CentOS 5.11, 6.9, 7.3 SUSE 11 SP4, 12 SP2 Windows Windows Server 2003 R2 SP2 Windows Server 2008 R1 SP2, 2008 R2 SP1 Windows Server 2012 R1, 2012 R2 Windows Server 2016

Setting Up AWS Application Discovery Service

Before you use AWS Application Discovery Service for the first time, complete the following tasks:

[Step 1: Sign Up for AWS \(p. 3\)](#)

[Step 2: Create an IAM User \(p. 3\)](#)

[Step 3: Attach Required IAM User Policies \(p. 4\)](#)

Step 1: Sign Up for AWS

When you sign up for Amazon Web Services (AWS), you are charged only for the services that you use. If you already have an AWS account, you can skip ahead to step 2. If you don't have an AWS account, use the following procedure to create one.

To create an AWS account

1. Open <https://aws.amazon.com/>, and then choose **Create an AWS Account**.

Note

This might be unavailable in your browser if you previously signed into the AWS Management Console. In that case, choose **Sign in to a different account**, and then choose **Create a new AWS account**.

2. Follow the online instructions.

Part of the sign-up procedure involves receiving a phone call and entering a PIN using the phone keypad.

Note your AWS account number, because you'll need it for the next task.

Step 2: Create an IAM User

Services such as AWS Application Discovery Service, require that you provide credentials when you access them. This way the service can determine whether you have permissions to access its resources. We recommend that you don't use the AWS account root user credentials to make requests. Instead, create an AWS Identity and Access Management (IAM) user, and grant that user full access. We refer to these users as having administrator-level credentials. You can use the administrator-level credentials to interact with AWS and perform tasks such as create an AWS S3 bucket, create additional IAM users, and grant permissions. For more information, see [Root Account Credentials vs. IAM User Credentials](#) in the *AWS General Reference* and [IAM Best Practices](#) in the *IAM User Guide*.

If you signed up for AWS but have not created an IAM user for yourself, you can create one using the IAM console.

To create an IAM user for yourself and add the user to an Administrators group

1. Use your AWS account email address and password to sign in as the *AWS account root user* to the IAM console at <https://console.aws.amazon.com/iam/>.

Note

We strongly recommend that you adhere to the best practice of using the **Administrator** IAM user below and securely lock away the root user credentials. Sign in as the root user only to perform a few [account and service management tasks](#).

2. In the navigation pane of the console, choose **Users**, and then choose **Add user**.
3. For **User name**, type **Administrator**.
4. Select the check box next to **AWS Management Console access**, select **Custom password**, and then type the new user's password in the text box. You can optionally select **Require password reset** to force the user to create a new password the next time the user signs in.
5. Choose **Next: Permissions**.
6. On the **Set permissions for user** page, choose **Add user to group**.
7. Choose **Create group**.
8. In the **Create group** dialog box, type **Administrators**.
9. For **Filter**, choose **Job function**.
10. In the policy list, select the check box for **AdministratorAccess**. Then choose **Create group**.
11. Back in the list of groups, select the check box for your new group. Choose **Refresh** if necessary to see the group in the list.
12. Choose **Next: Review** to see the list of group memberships to be added to the new user. When you are ready to proceed, choose **Create user**.

You can use this same process to create more groups and users, and to give your users access to your AWS account resources. To learn about using policies to restrict users' permissions to specific AWS resources, go to [Access Management](#) and [Example Policies](#).

To sign in as this new IAM user, first sign out of the AWS Management Console. Next, use the following URL, where *your_aws_account_id* is your AWS account number without the hyphens. For example, if your AWS account number is 1234-5678-9012, then *your_aws_account_id* is 123456789012).

```
https://your_aws_account_id.signin.aws.amazon.com/console/
```

Enter the IAM user name and password that you just created. When you're signed in, the navigation bar displays *your_user_name@your_aws_account_id*.

If you don't want the URL for your sign-in page to contain your AWS account ID, you can create an account alias. From the IAM dashboard, click **Create Account Alias** and enter an alias, such as your company name. To sign in after you create an account alias, use the following URL:

```
https://your_account_alias.signin.aws.amazon.com/console/
```

To verify the sign-in link for IAM users for your account, open the IAM console and check under **AWS Account Alias** on the dashboard.

Step 3: Attach Required IAM User Policies

Application Discovery Service uses the IAM-managed policies listed here to control access to the service or components of the service. For information about how to attach these managed policies to an IAM user account, see [Working with Managed Policies](#) in the *IAM User Guide*.

AWSApplicationDiscoveryServiceFullAccess

Grants the IAM user account access to the Application Discovery Service and Migration Hub APIs. With this policy, the user can configure Application Discovery Service, start and stop agents, start and stop agentless discovery, and query data from the AWS Discovery Service database.

AWSApplicationDiscoveryAgentAccess

Grants the Application Discovery Agent access to register and communicate with Application Discovery Service. Attach this policy to any user whose credentials are used by Application Discovery Agent. This policy also grants the user access to Arsenal. Arsenal is an agent service that is managed and hosted by AWS. Arsenal forwards data to Application Discovery Service in the cloud.

AWSAgentlessDiscoveryService

Grants the AWS Agentless Discovery Connector that is running in your VMware vCenter Server the access to register, communicate with, and share connector health metrics with Application Discovery Service. Attach this policy to any user whose credentials are used by the connector.

Each of the Application Discovery Service managed policies is shown here so that you can customize them as needed.

AWSApplicationDiscoveryServiceFullAccess

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "mgh:*",
        "discovery:*"
      ],
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Action": [
        "iam:GetRole"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

AWSApplicationDiscoveryAgentAccess

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "arsenal:RegisterOnPremisesAgent"
      ],
      "Resource": "*"
    }
  ]
}
```

AWSAgentlessDiscoveryService


```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "awsconnector:RegisterConnector",
        "awsconnector:GetConnectorHealth"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "iam:GetUser",
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:ListBucket"
      ],
      "Resource": [
        "arn:aws:s3:::connector-platform-upgrade-info/*",
        "arn:aws:s3:::connector-platform-upgrade-info",
        "arn:aws:s3:::connector-platform-upgrade-bundles/*",
        "arn:aws:s3:::connector-platform-upgrade-bundles",
        "arn:aws:s3:::connector-platform-release-notes/*",
        "arn:aws:s3:::connector-platform-release-notes",
        "arn:aws:s3:::prod.agentless.discovery.connector.upgrade/*",
        "arn:aws:s3:::prod.agentless.discovery.connector.upgrade"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:PutObjectAcl"
      ],
      "Resource": [
        "arn:aws:s3:::import-to-ec2-connector-debug-logs/*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "SNS:Publish"
      ],
      "Resource": "arn:aws:sns:*:*:metrics-sns-topic-for-*"
    },
    {
      "Sid": "Discovery",
      "Effect": "Allow",
      "Action": [
        "Discovery:*"
      ],
      "Resource": "*"
    },
    {
      "Sid": "arsenal",
      "Effect": "Allow",
      "Action": [
        "arsenal:RegisterOnPremisesAgent"
      ],
      "Resource": "*"
    }
  ]
}

```

```
}  
  ]  
}
```

Getting Started with AWS Application Discovery Service

In this section, you can find information about how to get started with AWS Application Discovery Service. Included are topics about how to access the Application Discovery Service console and the two ways of discovering data on your local servers.

Topics

- [Assumptions \(p. 8\)](#)
- [Accessing AWS Application Discovery Service \(p. 8\)](#)
- [Two Ways to Start Collecting Data \(p. 8\)](#)
- [AWS Agentless Discovery Connector \(p. 9\)](#)
- [AWS Application Discovery Agent \(p. 16\)](#)

Assumptions

To use Application Discovery Service, the following is assumed:

- You have signed up for AWS. For more information, see [Setting Up AWS Application Discovery Service \(p. 3\)](#)
- The Application Discovery Service tools only send their operational status if you have authorized, that is, connected, them.
- For a list of AWS Regions where you can use Application Discovery Service, see the [Amazon Web Services General Reference](#).

Accessing AWS Application Discovery Service

You can use AWS Application Discovery Service to help you plan your migration to the AWS Cloud by collecting usage and configuration data about your on-premises servers. Using the integrated AWS Migration Hub console, you can view the discovered servers, group them into applications, and then track the migration status of each application. You can find AWS Application Discovery Service at [AWS Application Discovery Service](#).

Additionally, you can use the AWS Application Discovery Service API to export system data and network connections about your discovered servers. This information helps you to group your servers into applications for migration planning. For more information about the API, see the [Application Discovery Service API Reference](#).

You can also use the AWS SDKs to develop applications that interact with Application Discovery Service. The AWS SDKs for Java, .NET, and PHP wrap the underlying Application Discovery Service API to simplify your programming tasks. For information about downloading the SDK libraries, see [Sample Code Libraries](#).

Two Ways to Start Collecting Data

To begin collecting data, use either the **Discovery Connector** or the **Discovery Agent**. A detailed description follows in each topic that will help you decide which discovery tool to use. Each of the following topics also has instructions to guide you through installation and deployment.

Topics

- [AWS Agentless Discovery Connector \(p. 9\)](#)
- [AWS Application Discovery Agent \(p. 16\)](#)

AWS Agentless Discovery Connector

Agentless discovery uses the AWS Discovery Connector. The AWS Discovery Connector is a VMware appliance that can collect information only about VMware virtual machines (VMs). This mode doesn't require you to install a connector on each host. You install the Discovery Connector as a VM in your VMware vCenter Server environment using an Open Virtualization Archive (OVA) file. Because the Discovery Connector relies on VMware metadata to gather server information regardless of operating system, it minimizes the time required for initial on-premises infrastructure assessment.

After you deploy and configure the Discovery Connector, it registers with the Application Discovery Service endpoint, <https://arsenal.us-west-2.amazonaws.com/>, and pings the service at regular intervals, approximately every 60 minutes, for configuration information. When you start the connector's data collecting process, it connects to VMware vCenter Server where it collects information about all the VMs and hosts managed by this specific vCenter. The collected data is sent to the Application Discovery Service using Secure Sockets Layer (SSL) encryption. The connector is configured to automatically upgrade when new versions of the connector become available. You can change this configuration setting at any time.

Topics

- [Data Collected by the Discovery Connector \(p. 9\)](#)
- [Download the Discovery Connector \(p. 12\)](#)
- [Deploy the Discovery Connector \(p. 12\)](#)
- [Configure the AWS Discovery Connector \(p. 13\)](#)
- [Start Discovery Connector Data Collection \(p. 15\)](#)

Data Collected by the Discovery Connector

The Discovery Connector collects information about your VMware vCenter Server hosts and VMs, including performance data about those hosts and VMs. However, you can capture this data only if VMware vCenter Server tools are installed. See [Step 3: Attach Required IAM User Policies \(p. 4\)](#) for Discovery Connector installation prerequisites.

Following, you can find an inventory of the information collected by the Discovery Connector.

Table legend for Discovery Connector collected data:

- Collected data is in measurements of kilobytes (KB) unless stated otherwise.
- Equivalent data in the Migration Hub console is reported in megabytes (MB).
- Data fields denoted with an asterisk (*) are only available in the .csv files produced from the connector's API export function.
- The polling period is in intervals of approximately 60 minutes.
- Data fields denoted with a double asterisk (**) currently return a *null* value.

Data field	Description
applicationConfigurationId*	ID of the migration application the VM is grouped under
avgCpuUsagePct	Average percentage of CPU usage over polling period
avgDiskBytesReadPerSecond	Average number of bytes read from disk over polling period
avgDiskBytesWrittenPerSecond	Average number of bytes written to disk over polling period
avgDiskReadOpsPerSecond**	Average number of read I/O operations per second null
avgDiskWriteOpsPerSecond**	Average number of write I/O operations per second
avgFreeRAM	Average free RAM expressed in MB
avgNetworkBytesReadPerSecond	Average amount of throughput of bytes read per second
avgNetworkBytesWrittenPerSecond	Average amount of throughput of bytes written per second
configId	Application Discovery Service assigned ID to the discovered VM
configType	Type of resource discovered
connectorId	ID of the Discovery Connector virtual appliance
cpuType	vCPU for a VM, actual model for a host
datacenterId	ID of the vCenter
hostId*	ID of the VM host
hostName	Name of host running the virtualization software
hypervisor	Type of hypervisor
id	ID of server
lastModifiedTimeStamp*	Latest date and time of data collection before data export
macAddress	MAC address of the VM
manufacturer	Maker of the virtualization software
maxCpuUsagePct	Max. percentage of CPU usage during polling period
maxDiskBytesReadPerSecond	Max. number of bytes read from disk over polling period

Data field	Description
maxDiskBytesWrittenPerSecond	Max. number of bytes written to disk over polling period
maxDiskReadOpsPerSecond**	Max. number of read I/O operations per second
maxDiskWriteOpsPerSecond**	Max. number of write I/O operations per second
maxNetworkBytesReadPerSecond	Max. amount of throughput of bytes read per second
maxNetworkBytesWrittenPerSecond	Max. amount of throughput of bytes written per second
memoryReservation*	Limit to avoid overcommitment of memory on VM
moRefId	Unique vCenter Managed Object Reference ID
name*	Name of VM or network (user specified)
numCores	Number of independent processing units within CPU
numCpus	Number of central processing units on VM
numDisks**	Number of disks on VM
numNetworkCards**	Number of network cards on VM
osName	Operating system name on VM
osVersion	Operating system version on VM
portGroupId*	ID of group of member ports of VLAN
portGroupName*	Name of group of member ports of VLAN
powerState*	Status of power
serverId	Application Discovery Service assigned ID to the discovered VM
smBiosId*	ID/version of the system management BIOS
state*	Status of the Discovery Connector virtual appliance
tagKey	User-defined key to store custom data or metadata about servers
tagValue	User-defined value to further define a key's custom data or metadata about servers
toolsStatus	Operational state of VMware tools (See Viewing and Sorting Data Collectors (p. 35) for a complete list.)
totalDiskSize	Total capacity of disk expressed in MB
totalRAM	Total amount of RAM available on VM in MB

Data field	Description
type	Type of host
vCenterId	Unique ID number of a VM
vCenterName*	Name of the vCenter host
virtualSwitchName*	Name of the virtual switch
vmFolderPath	Directory path of VM files
vmName	Name of the virtual machine

Download the Discovery Connector

Download, Set Up, and Start Collecting Data

To set up agentless discovery, you must download and deploy the Discovery Connector, which is a virtual appliance, on a VMware vCenter Server host in your on-premises environment. The Discovery Connector is an Open Virtualization Archive (OVA) file that you must install in your on-premises VMware environment.

Reminder

Discovery Connector supports VMware vCenter versions V5.5, V6, and V6.5.

Beginning with this section and those that follow on this page, you will be instructed how to download, deploy, configure, and start collecting data using the Discovery Connector.

To download the Discovery Connector OVA file and verify its checksum.

1. Sign in to vCenter as a VMware administrator and switch to the directory where you want to download the Discovery Connector OVA file.
2. Download the [Discovery Connector OVA](#).
3. Depending on which hashing algorithm you use in your system environment, download either the [MD5](#) or [SHA256](#) to get the file containing the checksum value. Use this value to verify the `AWSDiscoveryConnector.ova` file downloaded in the preceding step.
4. Depending on your variation of Linux, run the version appropriate MD5 command or SHA256 command to verify the cryptographic signature of the `AWSDiscoveryConnector.ova` file as shown following:

```
$ md5sum AWSDiscoveryConnector.ova
MD5 (AWSDiscoveryConnector.ova) = a5590986af27d8eabadf604cb1dbc13b
$ sha256sum AWSDiscoveryConnector.ova
SHA256(AWSDiscoveryConnector.ova)=
7cebe800a57aee863cab225abefdb454deeca998ef1defa00331a6c40d74d969
```

Verify that the checksum value returned from the command you ran is equal to the respective value displayed in the example above.

Deploy the Discovery Connector

Deploy the downloaded OVA file of the Discovery Connector in your VMware environment.

To deploy the Discovery Connector

1. Sign in to vCenter as a VMware administrator.
2. Choose **File, Deploy OVF Template** and select the ova file you downloaded in the previous section. Complete the wizard.
3. On the **Disk Format** page, select one of the thick provision disk types. We recommend that you choose **Thick Provision Eager Zeroed**, because it has the best performance and reliability. However, it requires several hours to zero out the disk. Do not choose **Thin Provision**. This option makes deployment faster but significantly reduces disk performance. For more information, see [Types of supported virtual disks](#) in the VMware documentation.
4. Locate and open the context (right-click) menu for the newly deployed template in the vSphere client inventory tree and choose **Power, Power On**.
5. Open the context (right-click) menu for the template again and choose **Open Console**. The console displays the IP address of the connector console. Make note of the IP address as you'll need it in order to complete the connector setup process.

Configure the AWS Discovery Connector

To finish the setup process, open a web browser and complete the following procedure and optional tasks within this section.

To configure the connector using the VMWare console

1. In a web browser, type the following URL in the address bar: **https://<ip_address>/**, where *ip_address* is the IP address of the connector console that you saved earlier.
2. Choose **Get started now** and follow the wizard steps.
3. In **Step 5: Discovery Connector Set Up** of the wizard steps, choose **Configure vCenter credentials**:
 - a. For **vCenter Host**, type the hostname or IP address of your VMware vCenter Server host.
 - b. For **vCenter Username**, type the name of a local or domain user that the connector uses to communicate with vCenter. For domain users, use the form *domain\username* or *username@domain*.
 - c. For **vCenter Password**, type the local or domain user password.
 - d. Choose **Ignore security certificate** to bypass SSL certificate validation with vCenter.
4. Choose **Configure AWS credentials** and type the credentials for the IAM user who is assigned the `AWSAgentlessDiscoveryService` IAM policy that you created in [Step 3: Attach Required IAM User Policies \(p. 4\)](#), and then choose **Next**.
5. Choose **Configure where to publish data** and select suitable publishing options. Choose **Next** and you should see the AWS Agentless Discovery Connector console.

Topics

- [Configure a static IP address for the connector \(p. 13\)](#)
- [Control the Scope of Data Collection \(p. 14\)](#)
- [Disabling Auto-Upgrades on AWS Discovery Connector \(p. 15\)](#)
- [Troubleshooting the Discovery Connector \(p. 15\)](#)

Configure a static IP address for the connector

This optional procedure is required if your environment requires that you use a static IP address.

To Configure a static IP address for the connector

1. Open the connector's virtual machine console and log in as **ec2-user** with the password **ec2pass**. Supply a new password if prompted.
2. Run the command **sudo setup.rb** and enter the password for *ec2-user* when prompted to display the configuration menu.
3. Enter **2** to select **Reconfigure network settings**. This displays current network information and a submenu for making changes to the network settings.
4. In the submenu generated from the previous step, enter **2** to select **Set up a static IP**. This will display a form to supply network settings:
 - For each field, provide an appropriate value and press Enter. You should see output similar to the following where *nnn.nnn.nnn.nnn* is populated with the address numbers you entered for each field:

```
Setting up static IP:
 1. Enter IP address: <nnn.nnn.nnn.nnn>
 2. Enter netmask: <nnn.nnn.nnn.nnn>
 3. Enter gateway: <nnn.nnn.nnn.nnn>
 4. Enter DNS 1: <nnn.nnn.nnn.nnn>
 5. Enter DNS 2: <nnn.nnn.nnn.nnn>

Static IP address configured.
```

Control the Scope of Data Collection

The vCenter user requires read-only permissions on each ESX host or VM to inventory using Application Discovery Service. Using the permission settings, you can control which hosts and VMs are included in the data collection. You can either allow all hosts and VMs under the current vCenter to be inventoried, or grant permissions on a case-by-case basis.

Note

As a security best practice, we recommend against granting additional, unneeded permissions to the vCenter user of the Discovery Connector.

The following procedures describe configuration scenarios ordered from least granular to most granular.

To discover data about *all* ESX hosts and VMs under the current vCenter

1. In your VMware vSphere client, choose **vCenter** and then choose either **Hosts and Clusters** or **VMs and Templates**.
2. Choose **Manage, Permissions**.
3. Select the vCenter user, open the context (right-click) menu, and choose **Change Role**.
4. In the **Assigned Role** pane, choose **Read-only**.
5. Choose **Propagate to children, OK**.

To discover data about a *specific* ESX host and *all* of its child objects

1. In your VMware vSphere client, choose **vCenter** and then choose either **Hosts and Clusters** or **VMs and Templates**.
2. Choose **Related Objects, Hosts**.
3. Open the context (right-click) menu for the host name and choose **All vCenter Actions, Add Permission**.
4. Under **Add Permission**, add the vCenter user to the host. For **Assigned Role**, choose **Read-only**.

5. Choose **Propagate to children, OK**.

Discover data about a *specific* ESX host or child VM

1. In your VMware vSphere client, choose **vCenter** and then choose either **Hosts and Clusters** or **VMs and Templates**.
2. Choose **Related Objects**.
3. Choose **Hosts** (showing a list of ESX hosts known to vCenter) or **Virtual Machines** (showing a list of VMs across all ESX hosts).
4. Open the context (right-click) menu for the host or VM name and choose **All vCenter Actions, Add Permission**.
5. Under **Add Permission**, add the vCenter user to the host or VM. For **Assigned Role**, choose **Read-only**.
6. Choose **OK**.

Note

If you chose **Propagate to children**, you can still remove the read-only permission from ESX hosts and VMs on a case-by-case basis. This option has no effect on inherited permissions applying to other ESX hosts and VMs.

Disabling Auto-Upgrades on AWS Discovery Connector

To ensure that you are running the latest version of AWS Discovery Connector, the auto-upgrade feature is enabled by default upon installation. However, you may disable the auto-upgrade feature as shown below.

To disable auto-upgrades

1. In a web browser, type the following URL in the address bar: **https://<ip_address>/**, where *ip_address* is the IP address of the AWS Discovery Connector.
2. In the Discovery Connector console, under **Actions**, choose **Disable Auto-Upgrade**.

Warning

Disabling auto-upgrades will prevent the latest security patches from being installed.

Troubleshooting the Discovery Connector

- The Discovery Connector does not support a standalone ESX host. The ESX host must be part of the vCenter Server instance.
- If you encounter problems and need help, contact [AWS Support](#). You will be contacted and may be asked to send the connector logs. To obtain the logs, do the following:
 - Log back in to the AWS Agentless Discovery Connector console (as you did during [configuration \(p. 13\)](#)) and choose **Download log bundle**.
 - Once the log bundle has finished downloading, send it as instructed by AWS Support.

Start Discovery Connector Data Collection

Now that you have deployed and configured the Discovery Connector in your VMware environment, you must complete the final step of actually turning on its data collection process. There are two ways to do this: Through the console or by making API calls through the AWS CLI. Instructions are provided below for both ways.

Start Data Collection Using the Migration Hub Console

You start the Discovery Connector data collection process on the **Data Collectors** page of the Migration Hub console.

To start data collection

1. In the navigation pane, choose **Data Collectors**.
2. Choose the **Connectors** tab.
3. Select the check box of the connector you want to start.
4. Choose **Start data collection**.

Note

If you don't see inventory information after starting data collection with the connector, confirm that you have registered the connector with your vCenter Server.

Start Data Collection Using the AWS CLI

To start the Discovery Connector data collection process from the AWS CLI, the AWS CLI must first be installed in your environment.

To install the AWS CLI and start data collection

1. Install the AWS CLI for your operating system (Linux, macOS, or Windows). See the [AWS Command Line Interface User Guide](#) for instructions.
2. Open the Command prompt (Windows) or Terminal (Linux or macOS).
 - a. Type `aws configure` and press Enter.
 - b. Enter your AWS Access Key Id and AWS Secret Access Key.
 - c. Enter `us-west-2` for the Default Region Name.
 - d. Enter `text` for Default Output Format.
3. Type the following command:

```
aws discovery start-data-collection-by-agent-ids --agent-ids <connector ID>
```

- If you don't know the ID of the connector you want to start, enter the following command exactly as shown to see the connector's ID:

```
aws discovery describe-agents --filters  
condition=EQUALS,name=hostName,values=connector
```

Note

If you don't see inventory information after starting data collection with the connector, confirm that you have registered the connector with your vCenter Server.

AWS Application Discovery Agent

The AWS Discovery Agent is AWS software that you install on on-premises servers and VMs targeted for discovery and migration. Agents capture system configuration, system performance, running processes, and details of the network connections between systems. Agents support most Linux and Windows operating systems, and you can deploy them on physical on-premises servers, Amazon EC2 instances, and virtual machines.

The Discovery Agent runs in your local environment and requires root privileges. When you start the Discovery Agent, it registers with the Application Discovery Service endpoint, *arsenal.aws.com*, and pings the service at 15 minute intervals for configuration information. When you send a command that tells an agent to start data collection, it starts collecting data for the host or VM where it resides. Collection includes system specifications, times series utilization or performance data, network connections, and process data. You can use this information to map your IT assets and their network dependencies. All of these data points can help you determine the cost of running these servers in AWS and also plan for migration.

Data is transmitted securely by the Discovery Agents to Application Discovery Service using Transport Layer Security (TLS) encryption. Agents are configured to upgrade automatically when new versions become available. You can change this configuration setting if desired.

Tip

Before downloading and beginning Discovery Agent installation, be sure to read through all of the required prerequisites in [Prerequisites for Agent Installation \(p. 19\)](#)

Topics

- [Data Collected by the Discovery Agent \(p. 17\)](#)
- [Prerequisites for Agent Installation \(p. 19\)](#)
- [Agent Installation on Linux \(p. 20\)](#)
- [Agent Installation on Windows \(p. 23\)](#)
- [Start Discovery Agent Data Collection \(p. 26\)](#)

Data Collected by the Discovery Agent

Following, you can find an inventory of the information collected by the Discovery Agent.

Table legend for Discovery Agent collected data:

- The term host refers to either a physical server or a VM.
- Collected data is in measurements of kilobytes (KB) unless stated otherwise.
- Equivalent data in the Migration Hub console is reported in megabytes (MB).
- Data fields denoted with an asterisk (*) are only available in the .csv files produced from the agent's API export function.
- The polling period is in intervals of approximately 15 minutes.

Data field	Description
agentAssignedProcessId*	Unique process ID of agent
agentId	Unique ID of agent
agentProvidedTimeStamp*	Date and time of agent observation (<i>mm/dd/yyyy hh:mm:ss am/pm</i>)
cmdLine*	Process entered at the command line
cpuType	Type of CPU (central processing unit) used in host
destinationIp*	IP address of device to which packet is being sent
destinationPort*	Port number to which the data/request is to be sent

Data field	Description
family*	Protocol of routing family
freeRAM (MB)	Free RAM expressed in MB
gateway*	Node address of network
hostName	Name of host data was collected on
hypervisor	Type of hypervisor
ipAddress	IP address of the host
ipVersion*	IP version number
isSystem*	Boolean attribute to indicate if a process is owned by the OS
macAddress	MAC address of the host
name*	Name of the host, network, metrics, etc. data is being collected for
netMask*	IP address prefix that a network host belongs to
osName	Operating system name on host
osVersion	Operating system version on host
path	Path of the command sourced from the command line
sourceIp*	IP address of the device sending the IP packet
sourcePort*	Port number from which the data/request originates from
timestamp*	Date and time of reported attribute logged by agent
totalCpuUsagePct	Percentage of CPU usage on host during polling period
totalDiskBytesReadPerSecond (Kbps)	Total amount of disk free space on host
totalDiskBytesWrittenPerSecond (Kbps)	Total size of disk on host
totalDiskFreeSize (GB)	Free disk space expressed in GB
totalDiskReadOpsPerSecond	Total number of read I/O operations per second
totalDiskSize (GB)	Total capacity of disk expressed in GB
totalDiskWriteOpsPerSecond	Total number of write I/O operations per second
totalNetworkBytesReadPerSecond (Kbps)	Total amount of throughput of bytes read per second
totalNetworkBytesWrittenPerSecond (Kbps)	Total amount of throughput of bytes written per second

Data field	Description
totalNumCores	Total number of independent processing units within CPU
totalNumCpus	Total number of central processing units
totalNumDisks	Total number of disks on host
totalNumLogicalProcessors*	Total number of physical cores times the number of threads that can run on each core
totalNumNetworkCards	Total count of network cards on server
totalRAM (MB)	Total amount of RAM available on host
transportProtocol*	Type of transport protocol used

Prerequisites for Agent Installation

These are the pre-installation tasks that should be performed to prevent errors from occurring during the actual installation of the agent. If you have a 1.x version of the agent installed, it needs to be removed before installing the latest version. Instructions for removing older versions are provided in the tasks below:

- Verify your OS environment is supported:

Linux

Amazon Linux 2012.03, 2015.03
Ubuntu 12.04, 14.04, 16.04
Red Hat Enterprise Linux 5.11, 6.9, 7.3
CentOS 5.11, 6.9, 7.3
SUSE 11 SP4, 12 SP2

Windows

Windows Server 2003 R2 SP2
Windows Server 2008 R1 SP2, 2008 R2 SP1
Windows Server 2012 R1, 2012 R2
Windows Server 2016

- If outbound connections from your network are restricted, you'll need to update your firewall settings. Agents require access to `arsenal` over TCP port 443 as in `https://arsenal.us-west-2.amazonaws.com:443`. They don't require any inbound ports to be open.
- Access to AWS S3 in us-west-2 is required for auto-upgrade to function.
- Create an IAM user with a policy providing agent access to Application Discovery Service. For information, see [Step 2: Create an IAM User \(p. 3\)](#).
- Check the time skew from your Network Time Protocol (NTP) servers and correct if necessary. Incorrect time skew causes the agent registration call to fail.
- Remove any previous-generation agents. If you previously installed Application Discovery Agent 1.0 for either Windows or Linux, you must uninstall it before continuing with the installation of the current agent.

Operating System	Command
Amazon Linux, Red Hat Enterprise Linux, CentOS	<code>yum remove AwsAgent</code>

Operating System	Command
Ubuntu Server	<code>apt-get remove awsagent</code>
Windows Server	Use Add/Remove Programs to uninstall AWS Agent .

Note

The Discovery Agent has a 32-bit agent executable, which works on both 32-bit and 64-bit operating systems. Having a single executable reduces the number of installation packages needed for deployment. This applies for both Linux and Windows OS and is addressed in their respective installation sections below.

Agent Installation on Linux

Complete the following procedure on Linux.

Note

If you are using a non-current Linux version, see [Requirements on Older Linux Platforms \(p. 21\)](#).

To install AWS Application Discovery Agent in your data center

1. Log in to your Linux-based server or VM and create a new directory to contain your agent components.
2. Switch to the new directory and download the installation script from either the command line or the console.
 - a. To download from the command line, run the following command.

```
curl -o ./aws-discovery-agent.tar.gz https://s3-us-west-2.amazonaws.com/aws-discovery-agent.us-west-2/linux/latest/aws-discovery-agent.tar.gz
```

- b. To download from the Migration Hub console, do the following:
 - i. Open the console and go to the [Discovery Tools](#) page.
 - ii. In the **Discovery Agent** box, choose **Download agent**, then choose **Linux** in the resultant list box. Your download begins immediately.
3. Verify the cryptographic signature of the installation package as shown following.

```
curl -o ./agent.sig https://s3-us-west-2.amazonaws.com/aws-discovery-agent.us-west-2/linux/latest/aws-discovery-agent.tar.gz.sigcurl -o ./discovery.gpg https://s3-us-west-2.amazonaws.com/aws-discovery-agent.us-west-2/linux/latest/discovery.gpggpg --no-default-keyring --keyring ./discovery.gpg --verify agent.sig aws-discovery-agent.tar.gz
```

The agent public key (discovery.gpg) fingerprint is 7638 F24C 6717 F97C 4F1B 3BC0 5133 255E 4DF4 2DA2.

4. Extract from the tarball as shown following.

```
tar -xzf aws-discovery-agent.tar.gz
```

5. Run the following command to install the agent in the us-west-2 Region.

```
sudo bash install -r us-west-2 -k <aws key id> -s <aws key secret>
```

Note

Agents automatically download and apply updates as they become available. We recommend using this default configuration. However, if you don't want agents to download and apply updates automatically, include the `-u false` parameter when running the installation script.

6. If outbound connections from your network are restricted, update your firewall settings. Agents require access to `arsenal` over TCP port 443 as in `us-west-2.amazonaws.com:443`. They don't require any inbound ports to be open.

Note

Agents also work with transparent web proxies. However, if you need to **configure a non-transparent proxy**, proceed to the next step.

7. Optional: To Configure a Non-Transparent Proxy:
 - a. Find the configuration file as described in [Agent Troubleshooting on Linux \(p. 23\)](#) and edit the file by adding the required configuration data as follows:

```
"proxyHost" : "<myproxy.mycompany.com>",  
"proxyPort" : <1234>,  
"proxyUser" : "<myusername>",  
"proxyPassword" : "<mypassword>",
```

- b. Save the edited configuration file ensuring that you still have valid json (taking care with the quotes and the commas). If your proxy doesn't require authentication, then leave out `proxyUser` and `proxyPassword`. While most proxies use http, if yours uses https, specify the following in the configuration file:

```
"proxyScheme" : "https"
```

- c. Restart the agent.

Note

If you encounter problems, add the following to the configuration file:

```
"enableAWSSDKLogging" : true
```

Then, restart the agent again, let it run for at least 15 minutes, and contact [AWS Support](#). They will help you troubleshoot and may ask you to send them the generated log files which can be found as described in [Agent Troubleshooting on Linux \(p. 23\)](#).

Topics

- [Requirements on Older Linux Platforms \(p. 21\)](#)
- [Manage the Discovery Agent Process on Linux \(p. 22\)](#)
- [Agent Troubleshooting on Linux \(p. 23\)](#)

Requirements on Older Linux Platforms

Some older Linux platforms such as SUSE 10, CentOS 5, and RHEL 5 are either at end of life or only minimally supported. These platforms can suffer from out-of-date cipher suites that prevent the agent installation script from downloading installation packages. They might also have a limited ability to find and download the platform libraries required by the agent from deprecated Linux repositories.

32-bit libc

One of the dependencies needed for the Application Discovery agent is 32-bit `libc`. This library must be installed on 64-bit systems that run the agent. If the installation script exits because it fails to find a suitable repository or otherwise fails to install 32-bit `libc`, you must manually find and install 32-bit `libc` before you can complete agent installation. Because 32-bit `libc` is a core Linux library, you must take great care in identifying a package that is compatible with your system. We recommend contacting AWS Support for assistance. After 32-bit `libc` is installed, run the installation script with the `-p false` parameter to skip the automated search of Linux repositories for prerequisites.

Curl

The Application Discovery agent requires `curl` for secure communications with the AWS server. Some old versions of `curl` are not able to communicate securely with a modern web service. To use the version of `curl` included with the Application Discovery agent for all operations, run the installation script with the `-c true` parameter.

Certificate Authority Bundle

Older Linux systems might have an out-of-date Certificate Authority (CA) bundle, which is critical to secure internet communication. To use the CA bundle included with the Application Discovery agent for all operations, run the installation script with the `-b true` parameter.

These three installation script options can be used in any combination. In the following example command, all three have been passed to the installation script:

```
sudo bash install -r us-west-2 -k <aws key id> -s <aws key secret> -p false -c true -b true
```

Manage the Discovery Agent Process on Linux

You can manage the behavior of the Discovery Agent at the system level using the following commands. For the correct command for each Linux distribution, see the following legend:

- (a) Amazon, Red Hat, or CentOS
- (b) Ubuntu
- (c) SUSE

Linux Commands for Application Discovery Agent

Task	Command
Verify that an agent is running	<pre>sudo systemctl status aws-discovery-daemon.service^a</pre> <pre>sudo initctl status aws-discovery-daemon^b</pre> <pre>sudo /etc/init.d/aws-discovery-daemon status^c</pre>
Start an agent	<pre>sudo systemctl start aws-discovery-daemon.service^a</pre> <pre>sudo initctl start aws-discovery-daemon^b</pre>

Task	Command
	<code>sudo /etc/init.d/aws-discovery-daemon start^c</code>
Stop an agent	<code>sudo systemctl stop aws-discovery-daemon.service^a</code> <code>sudo initctl stop aws-discovery-daemon^b</code> <code>sudo /etc/init.d/aws-discovery-daemon stop^c</code>
Restart an agent	<code>sudo systemctl restart aws-discovery-daemon.service^a</code> <code>sudo initctl restart aws-discovery-daemon^b</code> <code>sudo /etc/init.d/aws-discovery-daemon restart^c</code>
Uninstall an agent	<code>yum remove aws-discovery-agent^a</code> <code>apt-get remove aws-discovery-agent^b</code> <code>zypper remove aws-discovery-agent^c</code>

Agent Troubleshooting on Linux

If you encounter problems while installing or using the Application Discovery Agent on Linux, consult the following guidance about logging and configuration. When helping to troubleshoot potential issues with the agent or its connection to the Application Discovery Service, AWS Support often requests these files.

- **Log files**

Agent log files can be found under the following directory.

```
/var/log/aws/discovery/
```

Log files are named to indicate whether they are generated by the main daemon, the automatic upgrader, or installer.

- **Configuration files**

Agent configuration files can be found under the following directory.

```
/var/opt/aws/discovery/
```

- For instructions on how to remove older versions of the Discovery Agent, see [Prerequisites for Agent Installation](#) (p. 19).

Agent Installation on Windows

Complete the following procedure on Windows.

To install AWS Application Discovery Agent in your data center

1. Navigate to the [Microsoft Download Center](#) and choose **Download** to be taken to the download selection page, then on this page, select only `vc_redist.x86.exe` (do not select the "x64" version) regardless of the architecture of the machine you are installing on, then choose **Next**. Your download begins immediately.
2. Download the [Windows agent installer](#) but do not double-click and execute the installer within Windows.

Important

Do not double-click and execute the installer within Windows as it will fail to install. *Agent installation only works from the command prompt.* (If you already double-clicked on the installer, you must go to **Add/Remove Programs** and uninstall the agent before continuing on with the remaining installation steps.)

3. Open a command prompt as an administrator and navigate to the location where you saved the installation package.
4. To install the agent, run the following command.

```
msiexec.exe /i AWSDiscoveryAgentInstaller.msi REGION="us-west-2" KEY_ID="<aws key id>"  
KEY_SECRET="<aws key secret>" /q
```

Note

Agents automatically download and apply updates as they become available. We recommend this default configuration. To avoid downloading agents and applying updates automatically, include the following parameter when running the installation:
`AUTO_UPDATE=false`

Warning

Disabling auto-upgrades will prevent the latest security patches from being installed.

5. If outbound connections from your network are restricted, update your firewall settings. Agents require access to `arsenal` over TCP port 443 as in `us-west-2.amazonaws.com:443`. They do not require any inbound ports to be open.

Note

Agents also work with transparent web proxies. However, if you need to **configure a non-transparent proxy**, continue on with the following steps.

6. Optional: To Configure a Non-Transparent Proxy:
 - a. Find the configuration file as described in [Agent Troubleshooting on Windows \(p. 26\)](#), and edit the file by adding the required configuration data as follows:

```
"proxyHost" : "<myproxy.mycompany.com>",  
"proxyPort" : <1234>,  
"proxyUser" : "<myusername>",  
"proxyPassword" : "<mypassword>",
```

- b. Save the edited configuration file ensuring that you still have valid json (taking care with the quotes and the commas). If your proxy doesn't require authentication, then leave out `proxyUser` and `proxyPassword`. While most proxies use http, if yours uses https, specify the following in the configuration file:

```
"proxyScheme" : "https"
```

- c. Restart the agent.

Note

If you encounter problems, add the following to the configuration file:

```
"enableAWSSDKLogging" : true
```

Then, restart the agent again, let it run for 15 minutes, and troubleshoot what the issue may be by reading through the generated log files which can be found as described in [Agent Troubleshooting on Windows \(p. 26\)](#).

Topics

- [Package Signing on Windows 2003 \(p. 25\)](#)
- [Manage the Discovery Agent Process on Windows \(p. 25\)](#)
- [Agent Troubleshooting on Windows \(p. 26\)](#)

Package Signing on Windows 2003

For Windows Server 2008 and later, Amazon cryptographically signs the Application Discovery Service agent installation package with an SHA256 certificate. However, because the SHA2 certificate family is not supported by Windows Server 2003, the installation package for that platform is signed with an SHA1 certificate. Microsoft has published [hotfixes](#) that might allow your Windows 2003 systems to read an SHA256 certificate. If you require SHA256 in your Windows 2003 environment, contact AWS Support for assistance.

Manage the Discovery Agent Process on Windows

You can manage the behavior of the Discovery Agent at the system level through the Windows Server Manager Services console. The following table describes how.

Task	Service Name	Service Status/Action
Verify that an agent is running	AWS Discovery Agent AWS Discovery Updater	Started
Start an agent	AWS Discovery Agent AWS Discovery Updater	Choose Start
Stop an agent	AWS Discovery Agent AWS Discovery Updater	Choose Stop
Restart an agent	AWS Discovery Agent AWS Discovery Updater	Choose Restart

To uninstall a discovery agent on Windows

1. Open Control Panel in Windows.
2. Choose **Programs**.
3. Choose **Programs and Features**.
4. Select **AWS Discovery Agent**.
5. Choose **Uninstall**.

Agent Troubleshooting on Windows

If you encounter problems while installing or using the Application Discovery Agent on Windows, consult the following guidance about logging and configuration. When helping to troubleshoot potential issues with the agent or its connection to the Application Discovery Service, AWS Support often requests these files.

- **Installation logging**

In some cases, the `msiexec` command described preceding appears to fail. For example, a failure can appear with the Windows Services Manager showing that the discovery services are not being created. In this case, add `/L*V install.log` to the command to generate a verbose installation log.

- **Operational logging**

On Windows Server 2008 and later, agent log files can be found under the following directory.

```
C:\ProgramData\AWS\AWS Discovery\Logs
```

On Windows Server 2003, agent log files can be found under the following directory.

```
C:\Documents and Settings\All Users\Application Data\AWS\AWSDiscovery\Logs
```

Logs files are named to indicate whether generated by the main service, automatic upgrader, or installer.

- **Configuration file**

On Windows Server 2008 and later, the agent configuration file can be found at the following location.

```
C:\ProgramData\AWS\AWS Discovery\config
```

On Windows Server 2003, the agent configuration file can be found at the following location.

```
C:\Documents and Settings\All Users\Application Data\AWS\AWS Discovery\config
```

- For instructions on how to remove older versions of the Discovery Agent, see [Prerequisites for Agent Installation \(p. 19\)](#).

Start Discovery Agent Data Collection

Now that you have deployed and configured the Discovery Agent, you must complete the final step of actually turning on its data collection process. There are two ways to do this, through the console or by making API calls through the AWS CLI. Instructions are provided below for both ways by expanding your method of choice:

Start Data Collection Using the Migration Hub Console

You start the Discovery Agent data collection process on the **Data Collectors** page of the Migration Hub console.

To start data collection

1. In the navigation pane, choose **Data Collectors**.
2. Choose the **Agents** tab.
3. Select the check box of the agent you want to start.

Tip

If you installed multiple agents but only want to start data collection on certain hosts, the **Hostname** column in the agent's row identifies the host the agent is installed on.

4. Choose **Start data collection**.

Start Data Collection Using the AWS CLI

To start the Discovery Agent data collection process from the AWS CLI the AWS CLI must first be installed in your environment.

To install the AWS CLI and start data collection

1. If you have not already done so, install the AWS CLI appropriate to your OS type (Windows or Mac/Linux). See the [AWS Command Line Interface User Guide](#) for instructions.
2. Open the Command prompt (Windows) or Terminal (MAC/Linux).
 - a. Type `aws configure` and press Enter.
 - b. Enter your AWS Access Key Id and AWS Secret Access Key.
 - c. Enter `us-west-2` for the Default Region Name.
 - d. Enter `text` for Default Output Format.
3. Type the following command:

```
aws discovery start-data-collection-by-agent-ids --agent-ids <agent ID>
```

- If you don't know the ID of the agent you want to start, enter the following command to see the agent's ID:

```
aws discovery describe-agents
```

View and Export Discovered Data

The AWS Discovery Connector and AWS Discovery Agent both provide system performance data based on average and peak utilization. You can use the system performance data collected to perform a high-level TCO (total cost of ownership). Discovery Agents collect more detailed data including time series data for system performance information, inbound and outbound network connections, and processes running on the server. You can use this data to understand network dependencies between servers and group the related servers as applications for migration planning.

In this section you'll find instructions on how to view and work with data discovered by Discovery Connectors and Discovery Agents from both the console and the AWS CLI.

Topics

- [View Collected Data Using the Console \(p. 28\)](#)
- [Export Collected Data \(p. 28\)](#)
- [Explore Agent Collected Data in Athena \(p. 30\)](#)

View Collected Data Using the Console

After starting the data collection process of your Discovery Connector or Discovery Agent, you can use the console to view their collected data about your servers and VMs. Data appears in the console approximately 15 minutes after turning on data collection. This data can also be viewed in a csv format by exporting the collected data by making API calls through the AWS CLI. Exporting collected data is covered in the next section [Export Collected Data \(p. 28\)](#).

To view collected data about discovered servers

1. In the console's navigation pane, choose **Servers**. The discovered servers appear in the servers list.
2. For details comprised of the collected data, choose the server name link in the **Server info** column. Doing so displays a screen that describes detail information such as system information, performance metrics, and more.

To learn more about using the console to view, sort, and tag servers discovered by your Discovery Connectors or Discovery Agents, see [AWS Application Discovery Service Console Walkthroughs \(p. 32\)](#).

Export Collected Data

After starting the data collection process of your Discovery Connector or Discovery Agent, you can export their collected data about your servers and VMs. This data can be exported either by interacting with the console or by making API calls through the AWS CLI depending on which discovery tool you used to collect data.

- **Discovery Agent**, you can export the collected data either from the console or from the AWS CLI.
- **Discovery Connector**, you can only export the collected data from the AWS CLI.

Instructions are provided below for both ways by expanding your method of choice:

Export System Performance Data for All Servers

Collected data from all the Discovery Connectors and Discovery Agents running on your hosts and VMs can be bulk exported from the AWS CLI. If not already installed, the AWS CLI must first be installed in your environment.

To install the AWS CLI and export collected data

1. If you have not already done so, install the AWS CLI appropriate to your OS type (Windows or Mac/Linux). See the [AWS Command Line Interface User Guide](#) for instructions.
2. Open the Command prompt (Windows) or Terminal (MAC/Linux).
 - a. Type `aws configure` and press Enter.
 - b. Enter your AWS Access Key Id and AWS Secret Access Key.
 - c. Enter `us-west-2` for the Default Region Name.
 - d. Enter `text` for Default Output Format.
3. Type the following command to generate an export ID:

```
aws discovery start-export-task
```

4. Using the export ID generated in the previous step, type the following command to generate an S3 URL as a value for the parameter "configurationsDownloadUrl":

```
aws discovery describe-export-tasks --export-ids <export ID>
```

5. Copy the URL generated in the previous step and paste it in a browser to download the zip file with collected data of the discovered servers.

Export Agent Collected Data Using the Console

Exporting agent collected data from the console is limited to one agent when you are on the detail page for a specific server. There, you can find the server's export jobs listed at the bottom of the screen, underneath **Exports**. If no export jobs yet exist, the table is empty. You can execute up to five exports of server data at a time.

To export collected data about a discovered server

1. In the navigation pane, choose **Servers**.
2. In the **Server info** column, choose the link for the server that you want to export data for.
3. In the **Exports** section at the bottom of the screen, choose **Export server details**.
4. For **Export server details**, fill in **Start date** and **Time**.

Note

The start time can't be more than 72 hours prior from the current time.

5. Choose **Export** to start the job. The initial status is **In-progress**; to update the status, click the refresh icon for the **Exports** section.
6. When the export job is complete, choose **Download** and save the .zip file.
7. Unzip the saved file. A set of .csv files contains the export data, similar to the following:

- `<AWS account ID>_destinationProcessConnection.csv`
- `<AWS account ID>_networkInterface.csv`
- `<AWS account ID>_osInfo.csv`
- `<AWS account ID>_process.csv`

- `<AWS account ID>_sourceProcessConnection.csv`
- `<AWS account ID>_systemPerformance.csv`

You can open the .csv files in Microsoft Excel and review the exported server data.

Among the files, you can find a JSON file containing data about the export task and its results.

Explore Agent Collected Data in Athena

All the data collected by agents can also be exported and uploaded to your S3 bucket using the [AWS Discovery Utilities](#) scripts. These scripts enable bulk export of system performance, network connection, and process data for all the agent discovered servers, and transform it for use in [Amazon Athena](#) using an S3 bucket for storage. The AWS Discovery Utilities package includes the following scripts:

- **export.py** - performs a bulk export of data from all servers discovered by agents in CSV format.
- **convert_csv.py** - converts the CSV files to Parquet format and uploads them to your S3 bucket.
- **discovery_athena.ddl** - once the Parquet files are in S3, you modify the statements in `discovery_athena.ddl` to reference your S3 bucket. Then, run this script from the Athena console within a new or existing database.

After you have downloaded, modified, and run all of the above scripts, you can query the data in the Athena console using SQL commands. Instructions are provided below on how to do this along with sample SQL queries.

Note

- The **export.py** script will take longer to run if you have a large amount of discovered data.
- Before using Athena, please reference [Amazon Athena Pricing](#).

To explore collected data in Athena

1. Download the AWS Discovery Utilities scripts from [here](#).
2. Before running the scripts downloaded in the previous step, be sure to read the **README.md** file to ensure you understand the set up and have the correct environment configured, such as having *python* and *boto3* installed.
3. Run all three of the AWS Discovery Utilities scripts in the order specified and as instructed in the **README.md** file.

Note

Running the `convert_csv.py` script involves passing parameters, and you must modify the statements in `discovery_athena.ddl` to reference the correct S3 bucket before running it in the Athena console. All of this detail is provided in **README.md**.

4. After you have successfully run all of the scripts from the previous step, you are now ready to query the data in the Athena console using SQL commands.

Note

If you are not familiar with Amazon Athena, read [Get Started with Amazon Athena](#) before proceeding.

To analyze details about your on-premises servers, you can write your own SQL queries in Athena. Following are some example queries for reference.

- a. To obtain the network communication between servers based on a given port number, run the following query in the Athena Console:

```
WITH valid_ips AS
  (SELECT DISTINCT source_ip
   FROM source_process_connection ), outer_query AS
  (SELECT agent_id,
         source_ip,
         destination_ip,
         destination_port,
         count(*) AS frequency
   FROM source_process_connection
   WHERE ip_version = 'IPv4'
        AND destination_ip IN
        (SELECT *
         FROM valid_ips)
   GROUP BY agent_id, source_ip, destination_ip, destination_port )
SELECT source_ip AS Source,
       'Port ' || cast(destination_port AS varchar(20)) AS Edge, destination_ip
AS Target, Frequency
FROM outer_query;
```

- b. To obtain the system performance data for cost analysis, run the following query in the Athena Console:

```
SELECT DISTINCT SP.AGENT_ID,
               OS.OS_NAME,
               OS.OS_VERSION,
               MAX(SP.total_num_cores) AS Cores,
               MAX(SP.total_num_cpus) AS CPU,
               MAX(SP.total_disk_size_in_gb) AS StorageTotal,
               MAX(SP.total_disk_free_size_in_gb) AS StorageFree,
               MAX(SP.total_ram_in_mb) AS RAM,
               MAX(SP.total_disk_read_ops_per_sec) AS IOPS_Read,
               MAX(SP.total_disk_bytes_written_per_sec_in_kbps) AS IOPS_Write
FROM SYSTEM_PERFORMANCE AS SP, OS_INFO AS OS
WHERE SP.AGENT_ID = OS.AGENT_ID
GROUP BY SP.AGENT_ID, OS.OS_NAME, OS.OS_VERSION;
```

Removing your data from AWS Application Discovery Service

If you'd like to have all your data removed from Application Discovery Service, please contact [AWS Support](#) and request full data deletion from Application Discovery Service.

AWS Application Discovery Service Console Walkthroughs

AWS Application Discovery Service is integrated with AWS Migration Hub and customers can view and manage their data collectors, servers, and applications within Migration Hub. When you use the Application Discovery Service console, you are redirected to the Migration Hub console. Working with the Migration Hub console requires no extra steps or setup on your part.

In this section, you can find how to manage and monitor your Discovery Connectors and Discovery Agents using the console.

Topics

- [Main Dashboard \(p. 32\)](#)
- [Data Collection Tools \(p. 34\)](#)
- [View & Export Server Data \(p. 36\)](#)
- [Applications \(p. 38\)](#)

Main Dashboard

The main dashboard is selected by default on the homepage of the AWS Migration Hub console, or, by choosing **Dashboard** in the navigation pane. In Migration Hub's main dashboard, you can view high-level statistics about servers, applications, and data collectors such as Discovery Connectors and Discovery Agents.

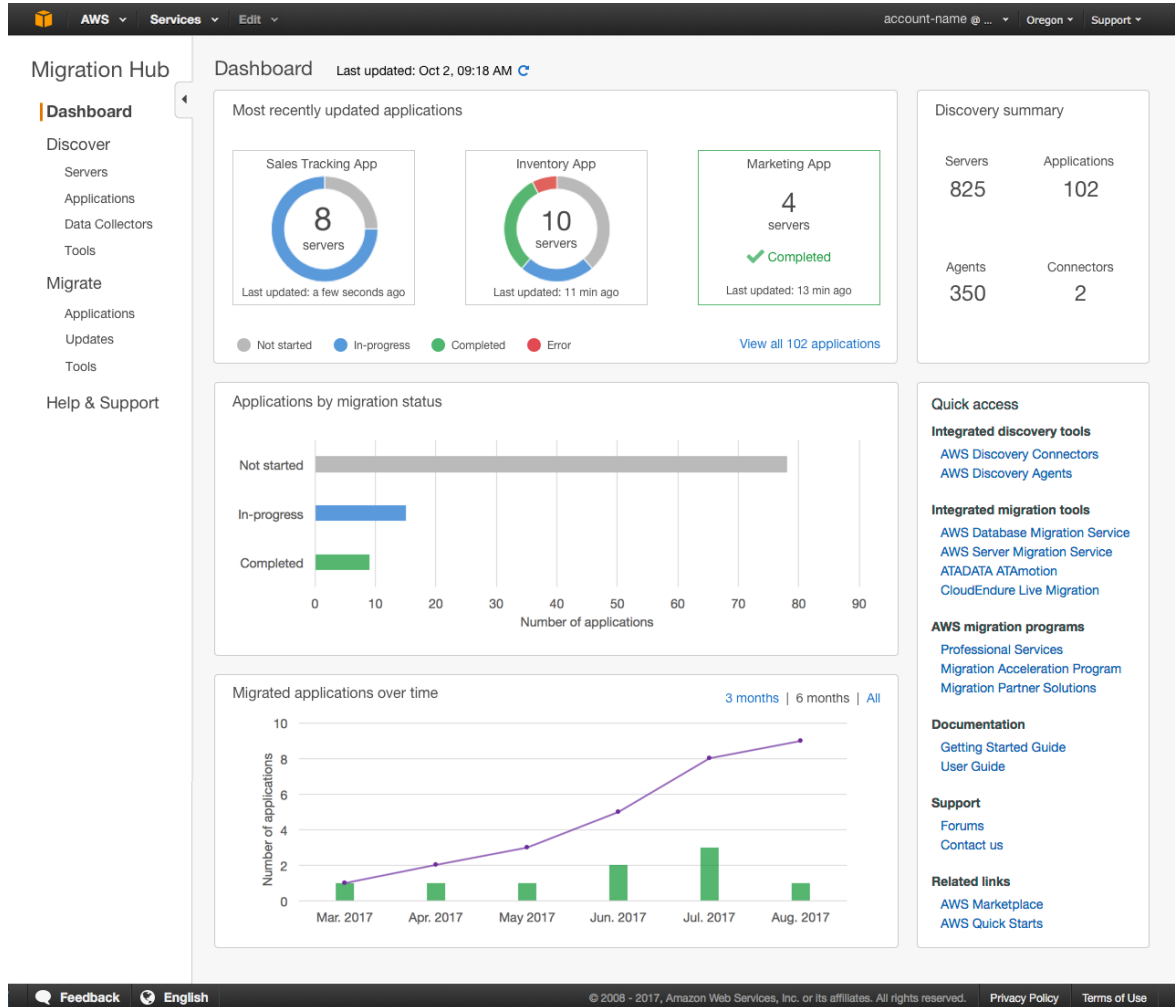
Topics

- [Main Dashboard \(p. 32\)](#)
- [Navigating from the Dashboard and the Navigation Pane \(p. 33\)](#)

Main Dashboard

The main dashboard gathers data from the **Discover** and **Migrate** dashboards in a central location. It has four status and information panes and a list of links for quick access. Using the panes, you can see a summary status of your most recently updated applications. You can also get quick access to any of your applications, get an overview of applications in different states, and track the migration progress over time.

To reach the main dashboard, choose **Dashboard** from the navigation pane, which is on the left side of the Migration Hub homepage.



Navigating from the Dashboard and the Navigation Pane

After you view dashboard data summaries, you might want to retrieve more detail. To do this, navigate directly from the relevant status or information box on the main dashboard.

In the table following, you can find instructions on how to navigate from a dashboard to the information you want to see. You can also find instructions on how to get to this information by using the navigation pane, which is on the left side of the Migration Hub homepage.

To See	Do This in the Dashboard	Do This in the Navigation Pane
All servers	In the main dashboard, in the Discovery summary box, choose Servers .	1. In the navigation pane, choose Servers .
All agents	In the main dashboard, in the Discovery summary box, choose Agents .	1. In the navigation pane, choose Data Collectors . 2. Choose the Agents tab.

To See	Do This in the Dashboard	Do This in the Navigation Pane
All connectors	In the main dashboard, in the Discovery summary box, choose Connectors .	<ol style="list-style-type: none"> 1. In the navigation pane, choose Data Collectors. 2. Choose the Connectors tab.
All applications	<p>From either the main dashboard or Migrate dashboard, in the Most recently updated applications box, choose View all applications.</p> <p>In the Discover dashboard in the Servers & Applications box, choose View all applications.</p>	<ol style="list-style-type: none"> 1. In the navigation pane, under Migrate, choose Applications. 2. Choose Applications.
Application details: <ul style="list-style-type: none"> • Migration status • Server list 	From either the main dashboard or Migrate dashboard, in the Most recently updated applications box, choose the application's status box.	<ol style="list-style-type: none"> 1. In the navigation pane, choose Migrate. 2. Choose Applications. 3. In the Application Name column, choose the application name.
Server details: <ul style="list-style-type: none"> • Basic information • Performance information 	<ol style="list-style-type: none"> 1. From either the main dashboard or Migrate dashboard, in the Most recently updated applications box, choose the application. 2. In the Server ID column, choose the server name . 	<ol style="list-style-type: none"> 1. In the navigation pane, choose Servers. 2. In the Server ID column, choose the server name.

Data Collection Tools

The Discovery Connector and Discovery Agent are the data collection tools that Application Discovery Service uses to help you discover your existing infrastructure. You can download and deploy discovery connectors and discovery agents as explained in [AWS Agentless Discovery Connector \(p. 9\)](#) and [AWS Application Discovery Agent \(p. 16\)](#).

These data collection tools store their data in the Application Discovery Service's repository, providing details about each server and the processes running on them. When either of these tools is deployed, you can start, stop, and view the collected data from the Migration Hub console.

Topics

- [Starting and Stopping Data Collectors \(p. 34\)](#)
- [Viewing and Sorting Data Collectors \(p. 35\)](#)

Starting and Stopping Data Collectors

Whether you deployed a Discovery Connector or a Discovery Agent, you can start or stop their data collection process on the **Data Collectors** page of the Migration Hub console.

To start or stop data collection tools

1. In the navigation pane, choose **Data Collectors**.
2. Choose either the **Connectors** or **Agents** tab.

3. Select the check box of the collection tool you want to start or stop.
4. Choose **Start data collection** or **Stop data collection**.

Viewing and Sorting Data Collectors

If you deployed many data collectors, you can sort the Discovery Connectors or Discovery Agents that are returned to the **Data Collectors** page of the console. You can do this by applying filters in the search bar. You can search and filter on most of the criteria specified in the **Data Collectors** list.

The following table shows the search criteria that you can use, including operators, values, and a definition of the values.

Search Criterion	Operator	Value: Definition
Collection status		<p>Started: Data is being collected and sent to Application Discovery Service.</p> <p>Start scheduled: Data collection is scheduled to start. Data will be sent on next ping, and status will change to Started.</p> <p>Stopped: Data is not being collected or sent to Application Discovery Service.</p> <p>Stop scheduled: Data collection is scheduled to stop. Data will stop being sent to Application Discovery Service on next ping, and status will change to Stopped.</p>
Health	<p>==</p> <p>!=</p>	<p>Healthy: Data collection isn't turned on. The tool is functioning normally.</p> <p>Unhealthy: The tool is in an error state. Data isn't being collected or sent to Application Discovery Service.</p> <p>Unknown: No connection established in over an hour.</p> <p>Shutdown: The tool last communicated "shutting down" due to a system restart. If a reboot or tool upgrade occurred, status will change to another state.</p> <p>Running: Data collection is turned on. The tool is functioning normally.</p>
Hostname		<p>For agents, any host name selected from the pre-populated list of hostnames.</p> <p>For connectors, not applicable.</p>
IP address		Any IP address selected from the pre-populated list where a collector is running.
Connector/Agent ID	==	Any connector or agent ID selected from the pre-populated list from the console.

To sort data collectors by applying search filters

1. In the navigation pane, choose **Data Collectors**.
2. Choose either the **Connectors** or **Agents** tab.
3. Click inside the search bar and choose a search criterion from the list.
4. Choose an operator from the next list.
5. Choose a value from the last list.

View & Export Server Data

The **Servers** page provides system configuration and performance data about each server instance known to the data collection tools. You can view server information, sort servers with filters, tag servers with key-value pairs, and export detailed server and system information.

Topics

- [Viewing and Sorting Servers \(p. 36\)](#)
- [Tagging Servers \(p. 36\)](#)
- [Exporting Server Data \(p. 37\)](#)

Viewing and Sorting Servers

You can view information about the servers discovered by the data collection tools, and you can sort through the servers using filters.

Viewing Servers

You can get a general view and a detailed view of the servers discovered by the data collection tools.

To view discovered servers

1. In the navigation pane, choose **Servers**. The discovered servers appear in the servers list.
2. For more detail about a server, choose its server link in the **Server info** column. Doing so displays a screen that describes the server.

The server's detail screen displays system information and performance metrics. You can also find a button to export network dependencies and processes information. To export detailed server information, see [Exporting Server Data \(p. 37\)](#).

Sorting Servers with Search Filters

To easily find specific servers, apply search filters to sort through all the servers discovered by the collection tools. You can search and filter on numerous criteria.

To sort servers by applying search filters

1. In the navigation pane, choose **Servers**.
2. Click inside the search bar, and choose a search criterion from the list.
3. Choose an operator from the next list.
4. Type in a case-sensitive value for the search criterion you selected, and press Enter.
5. Multiple filters can be applied by repeating steps 2 - 4.

Tagging Servers

To assist migration planning and help stay organized, you can create multiple tags for each server. *Tags* are user-defined key-value pairs that can store any custom data or metadata about servers. You can tag an individual server or multiple servers in a single operation. Application Discovery Service tags are similar to AWS tags, but the two types of tag cannot be used interchangeably.

You can add or remove multiple tags for one or more servers from the main **Servers** page. On a server's detail page, you can add or remove one or more tags for the selected server. You can do any type of tagging task involving multiple servers or tags in a single operation. You can also remove tags.

To add tags to one or more servers

1. In the navigation pane, choose **Servers**.
2. In the **Server info** column, choose the server link for the server that you want to add tags for. To add tags to more than one server at a time, click inside the check boxes of multiple servers.
3. Choose **Add tag**.
4. In the dialog box, type a value in the **Key** field, and optionally a value in the **Value** field.

Add more tags by choosing **Additional tag** and adding more information.

5. Choose **Add Tags**. A green confirmation message will be displayed at the top of the screen.
6. Optionally, tags can be added for an individual server from its detail page by choosing **Actions**, and then **Add tag** and repeating the above steps.

To remove tags from one or more servers

1. In the navigation pane, choose **Servers**.
2. In the **Server info** column, choose the server link for the server that you want to remove tags from. Click inside the check boxes of multiple servers to remove tags from more than one server at a time.
3. For **Actions**, choose **Remove tag**.
4. Select each tag you want to remove, or choose **select all**.
5. Choose **Remove**. A green confirmation message appears at the top of the screen.
6. Optionally, tags can be removed for an individual server from its detail page by choosing **Actions**, and then **Remove tag** and repeating the above steps.

Exporting Server Data

To export network dependencies and process information for one server at a time, you can use a server's detail screen. You can find the export jobs for a server in a table located in the **Exports** section of the server's detail screen. If no export jobs yet exist, the table is empty. You can simultaneously export up to five collections of data.

Note

Exporting server data from the console is only available for data collected by an agent running on that server. If you want to download data collected by a connector, see [Export System Performance Data for All Servers \(p. 29\)](#). Or, if you want to bulk export data for all servers where agents have been installed, see [Explore Agent Collected Data in Athena \(p. 30\)](#).

To export detailed server data

1. In the navigation pane, choose **Servers**.
2. In the **Server info** column, choose the ID of the server for which you want to export data.
3. In the **Exports** section at the bottom of the screen, choose **Export server details**.
4. For **Export server details**, fill in **Start date** and **Time**.

Note

The start time can't be more than 72 hours before the current time.

5. Choose **Export** to start the job. The initial status is **In-progress**; to update the status, click the refresh icon for the **Exports** section.
6. When the export job is complete, choose **Download** and save the .zip file.

7. Unzip the saved file. A set of .csv files contains the export data, similar to the following:

- <AWS account ID>_destinationProcessConnection.csv
- <AWS account ID>_networkInterface.csv
- <AWS account ID>_osInfo.csv
- <AWS account ID>_process.csv
- <AWS account ID>_sourceProcessConnection.csv
- <AWS account ID>_systemPerformance.csv

You can open the .csv files in Microsoft Excel and review the exported server data.

Among the files, you can find a JSON file containing data about the export task and its results.

Applications

Some of your discovered servers might need to be migrated together to remain functional. In this case, you can logically define and group discovered servers into applications.

As part of the grouping process, you can search, filter, and add tags.

To group servers into a new or existing application

1. In the navigation pane, choose **Servers**.
2. In the servers list, select each server that you want to group into a new or existing application.

To help choose servers for your group, you can search and filter on any criteria that you specify in the server list. Click inside the search bar and choose an item from the list, choose an operator from the next list, and then type in your criteria.

3. Optional: For each selected server, choose **Add tag**, type a value for **Key**, and then optionally type a value for **Value**.
4. Choose **Group as application** to create your application, or add to an existing one.
5. In the **Group as application** dialog box, choose **Group as a new application** or **Add to an existing application**.
 - a. If you chose **Group as a new application**, type a name for **Application name**. Optionally, you can type a description for **Application description**.
 - b. If you chose **Add to an existing application**, select the name of the application to add to in the list.
6. Choose **Save**.

AWS Application Discovery Service Service Limits

Application Discovery Service has the following limitations for agentless and agent-based discovery.

Agentless discovery

The service limits you to 10 GB of data per day. If you reach this limit, the service doesn't process any more data for that day. If you frequently reach this limit, contact [AWS Support](#) about extending the limit.

Agent-based discovery

Agent-based discovery currently enforces the following limits:

- 1,000 active agents — agents that are collecting and sending data to Application Discovery Service in the cloud.
- 10,000 inactive agents — agents that are responsive but not collecting data.
- 10 GB of data per day — collected by all agents associated with a given AWS account.
- 90 days of data storage — after which the data is purged.

Document History for AWS Application Discovery Service

The following table describes the important changes to the documentation since the last release of Application Discovery Service.

Latest documentation update: May 25, 2018

Change	Description	Date
Major revision	Rewrites to usage & output details; entire document restructured.	May 25, 2018
Discovery Agent 2.0	A new and improved Application Discovery agent was released.	October 19, 2017
Console	The AWS Management Console was added.	December 19, 2016
Agentless discovery	This release describes how to set up and configure agentless discovery.	July 28, 2016
New details for Microsoft Windows Server and command issue fixes	This update adds details about Microsoft Windows Server. It also documents fixes to various command issues.	May 20, 2016
Initial publication	This is the first release of the <i>Application Discovery Service User Guide</i> .	May 12, 2016

AWS Glossary

For the latest AWS terminology, see the [AWS Glossary](#) in the *AWS General Reference*.