



Set up a Connector

Cloud Manager

NetApp

March 15, 2022

This PDF was generated from https://docs.netapp.com/us-en/occm/concept_connectors.html on March 15, 2022. Always check docs.netapp.com for the latest.

Table of Contents

- Set up a Connector 1
 - Learn about Connectors..... 1
 - Networking requirements for the Connector 5
 - Creating a Connector in AWS from Cloud Manager 10
 - Creating a Connector in Azure from Cloud Manager..... 13
 - Creating a Connector in GCP from Cloud Manager..... 23
 - Install on your own host 30
 - Create from the marketplace 37

Set up a Connector

Learn about Connectors

In most cases, an Account Admin will need to deploy a *Connector* in your cloud or on-premises network. The Connector is a crucial component for the day-to-day use of Cloud Manager. The Connector enables Cloud Manager to manage the resources and processes within your public cloud environment.

When a Connector is required

A Connector is required to use any of the following services within Cloud Manager:

- Amazon FSx for ONTAP management features
- Amazon S3 bucket discovery
- Cloud Backup
- Cloud Data Sense
- Cloud Tiering
- Cloud Volumes ONTAP
- Global File Cache
- Kubernetes clusters
- Monitoring
- On-premises ONTAP clusters

A Connector is **not** required for the following services:

- Active IQ Digital Advisor
- Amazon FSx for ONTAP working environment creation

A Connector is required to create and manage volumes, replicate data, and integrate FSx for ONTAP with NetApp cloud services, such as Data Sense and Cloud Sync.

- Azure NetApp Files
- Cloud Volumes Service for Google Cloud
- Cloud Sync



While a Connector isn't required to set up and manage Azure NetApp Files, a Connector is required if you want to use Cloud Data Sense to scan Azure NetApp Files data.

Supported locations

A Connector is supported in the following locations:

- Amazon Web Services
- Microsoft Azure

- Google Cloud
- On your premises
- On your premises, without internet access

Note about Azure deployments

If you deploy the Connector in Azure, it should be deployed in the same Azure region as the Cloud Volumes ONTAP systems that it manages, or in the [Azure region pair](#) for the Cloud Volumes ONTAP systems. This requirement ensures that an Azure Private Link connection is used between Cloud Volumes ONTAP and its associated storage accounts. [Learn how Cloud Volumes ONTAP uses an Azure Private Link.](#)

Note about Google Cloud deployments

If you want to create a Cloud Volumes ONTAP system in Google Cloud, then you must have a Connector that's running in Google Cloud as well. You can't use a Connector that's running in AWS, Azure, or on-prem.

Shared Linux hosts are not supported

The Connector isn't supported on a VM that is shared with other applications. The VM must be dedicated to the Connector software.

3rd-party agents and extensions

3rd-party agents or VM extensions are not supported on the Connector VM.

Connectors should remain running

A Connector should remain running at all times. It's important for the continued health and operation of the services that you enable.

For example, a Connector is a key component in the health and operation of Cloud Volumes ONTAP PAYGO systems. If a Connector is powered down, Cloud Volumes ONTAP PAYGO systems will shut down after losing communication with a Connector for longer than 14 days.

How to create a Connector

An Account Admin needs to create a Connector before a Workspace Admin can create a Cloud Volumes ONTAP working environment and use any of the other features listed above.

An Account Admin can create a Connector in a number of ways:

- Directly from Cloud Manager (recommended)
 - [Create in AWS](#)
 - [Create in Azure](#)
 - [Create in GCP](#)
- By manually installing the software on your own Linux host
 - [On a host that has internet access](#)
 - [On an on-prem host that doesn't have internet access](#)
- From your cloud provider's marketplace

- [AWS Marketplace](#)
- [Azure Marketplace](#)

Cloud Manager will prompt you to create a Connector if one is needed to complete an action.

Permissions

Specific permissions are needed to create the Connector and another set of permissions are needed for the Connector instance itself.

Permissions to create a Connector

The user who creates a Connector from Cloud Manager needs specific permissions to deploy the instance in your cloud provider of choice. Cloud Manager will remind you of the permissions requirements when you create a Connector.

[View policies for each cloud provider.](#)

Permissions for the Connector instance

The Connector needs specific cloud provider permissions to perform operations on your behalf. For example, to deploy and manage Cloud Volumes ONTAP.

When you create a Connector directly from Cloud Manager, Cloud Manager creates the Connector with the permissions that it needs. There's nothing that you need to do.

If you create the Connector yourself from the AWS Marketplace, the Azure Marketplace, or by manually installing the software, then you'll need to make sure that the right permissions are in place.

- [View policies for each cloud provider](#)
- [Learn how Cloud Manager uses cloud provider permissions](#)



After you create a Connector in AWS, you can add an additional set of AWS credentials to enable the Connector to deploy and manage Cloud Volumes ONTAP in other AWS accounts. [Learn how to manage AWS credentials and subscriptions.](#)

Number of working environments per Connector

A Connector can manage multiple working environments in Cloud Manager. The maximum number of working environments that a single Connector should manage varies. It depends on the type of working environments, the number of volumes, the amount of capacity being managed, and the number of users.

If you have a large-scale deployment, work with your NetApp representative to size your environment. If you experience any issues along the way, reach out to us by using the in-product chat.

When to use multiple Connectors

In some cases, you might only need one Connector, but you might find yourself needing two or more Connectors.

Here are a few examples:

- You're using a multi-cloud environment (AWS and Azure), so you have one Connector in AWS and another in Azure. Each manages the Cloud Volumes ONTAP systems running in those environments.
- A service provider might use one NetApp account to provide services for their customers, while using another account to provide disaster recovery for one of their business units. Each account would have separate Connectors.

Using multiple Connectors with the same working environment

You can manage a working environment with multiple Connectors at the same time for disaster recovery purposes. If one Connector goes down, you can switch to the other Connector to immediately manage the working environment.

To set up this configuration:

1. [Switch to another Connector](#)
2. Discover the existing working environment.
 - [Adding existing Cloud Volumes ONTAP systems to Cloud Manager](#)
 - [Discovering ONTAP clusters](#)
3. Set the [Capacity Management Mode](#) to **Manual** on any additional Connectors.

Only the main Connector should be set to **Automatic Mode**. If you switch to another Connector for DR purposes, then you can change the Capacity Management Mode as needed.

When to switch between Connectors

When you create your first Connector, Cloud Manager automatically uses that Connector for each additional working environment that you create. Once you create an additional Connector, you'll need to switch between them to see the working environments that are specific to each Connector.

[Learn how to switch between Connectors.](#)

The local user interface

While you should perform almost all tasks from the [SaaS user interface](#), a local user interface is still available on the Connector. This interface is needed if you install the Connector in an environment that doesn't have internet access, and for a few tasks that need to be performed from the Connector itself, instead of the SaaS interface:

- [Setting a proxy server](#)
- Installing a patch (you'll typically work with NetApp personnel to install a patch)
- Downloading AutoSupport messages (usually directed by NetApp personnel when you have issues)

[Learn how to access the local UI.](#)

Connector upgrades

The Connector automatically updates its software to the latest version, as long as it has [outbound internet access](#) to obtain the software update.

Networking requirements for the Connector

Set up your networking so the Connector can manage resources and processes within your public cloud environment. The most important step is ensuring outbound internet access to various endpoints.

The information on this page is for a typical deployment where the Connector has outbound internet access.



If your network uses a proxy server for all communication to the internet, you can specify the proxy server from the Settings page. Refer to [Configuring the Connector to use a proxy server](#).

Connection to target networks

A Connector requires a network connection to the type of working environment that you're creating and the services that you're planning to enable.

For example, if you install a Connector in your corporate network, then you must set up a VPN connection to the VPC or VNet in which you launch Cloud Volumes ONTAP.

Possible conflict with IP addresses in the 172 range

If your network has a subnet configured in the 172 range, then you might experience connectivity failures from Cloud Manager. [Learn more about this known issue](#).

Outbound internet access

Outbound internet access is required from the Connector.

Endpoints to manage resources in your public cloud environment

The Connector requires outbound internet access to manage resources and processes within your public cloud environment.

Endpoints	Purpose
https://support.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support.
https://*.cloudmanager.cloud.netapp.com	To provide SaaS features and services within Cloud Manager.
https://cloudmanagerinfraprod.azurecr.io https://*.blob.core.windows.net	To upgrade the Connector and its Docker components.

Endpoints to install the Connector on a Linux host

You have the option to manually install the Connector software on your own Linux host. If you do, the installer for the Connector must access the following URLs during the installation process:

- <https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm>
- <https://s3.amazonaws.com/aws-cli/awscli-bundle.zip>

- https://*.blob.core.windows.net or <https://hub.docker.com>

The host might try to update operating system packages during installation. The host can contact different mirroring sites for these OS packages.

Ports and security groups

There's no incoming traffic to the Connector, unless you initiate it. HTTP and HTTPS provide access to the [local UI](#), which you'll use in rare circumstances. SSH is only needed if you need to connect to the host for troubleshooting.

Rules for the Connector in AWS

The security group for the Connector requires both inbound and outbound rules.

Inbound rules

Protocol	Port	Purpose
SSH	22	Provides SSH access to the Connector host
HTTP	80	Provides HTTP access from client web browsers to the local user interface and connections from Cloud Data Sense
HTTPS	443	Provides HTTPS access from client web browsers to the local user interface
TCP	3128	Provides the Cloud Data Sense instance with internet access, if your AWS network doesn't use a NAT or proxy
TCP	9060	Provides the ability to enable and use Cloud Data Sense (required only for GovCloud deployments)

Outbound rules

The predefined security group for the Connector opens all outbound traffic. If that is acceptable, follow the basic outbound rules. If you need more rigid rules, use the advanced outbound rules.

Basic outbound rules

The predefined security group for the Connector includes the following outbound rules.

Protocol	Port	Purpose
All TCP	All	All outbound traffic
All UDP	All	All outbound traffic

Advanced outbound rules

If you need rigid rules for outbound traffic, you can use the following information to open only those ports that are required for outbound communication by the Connector.



The source IP address is the Connector host.

Service	Protocol	Port	Destination	Purpose
Active Directory	TCP	88	Active Directory forest	Kerberos V authentication
	TCP	139	Active Directory forest	NetBIOS service session
	TCP	389	Active Directory forest	LDAP
	TCP	445	Active Directory forest	Microsoft SMB/CIFS over TCP with NetBIOS framing
	TCP	464	Active Directory forest	Kerberos V change & set password (SET_CHANGE)
	TCP	749	Active Directory forest	Active Directory Kerberos V change & set password (RPCSEC_GSS)
	UDP	137	Active Directory forest	NetBIOS name service
	UDP	138	Active Directory forest	NetBIOS datagram service
	UDP	464	Active Directory forest	Kerberos key administration
API calls and AutoSupport	HTTPS	443	Outbound internet and ONTAP cluster management LIF	API calls to AWS and ONTAP, and sending AutoSupport messages to NetApp
API calls	TCP	3000	ONTAP HA mediator	Communication with the ONTAP HA mediator
	TCP	8088	Backup to S3	API calls to Backup to S3
DNS	UDP	53	DNS	Used for DNS resolve by Cloud Manager
Cloud Data Sense	HTTP	80	Cloud Data Sense instance	Cloud Data Sense for Cloud Volumes ONTAP

Rules for the Connector in Azure

The security group for the Connector requires both inbound and outbound rules.

Inbound rules

Port	Protocol	Purpose
22	SSH	Provides SSH access to the Connector host
80	HTTP	Provides HTTP access from client web browsers to the local user interface
443	HTTPS	Provides HTTPS access from client web browsers to the local user interface

Outbound rules

The predefined security group for the Connector opens all outbound traffic. If that is acceptable, follow the basic outbound rules. If you need more rigid rules, use the advanced outbound rules.

Basic outbound rules

The predefined security group for the Connector includes the following outbound rules.

Port	Protocol	Purpose
All	All TCP	All outbound traffic
All	All UDP	All outbound traffic

Advanced outbound rules

If you need rigid rules for outbound traffic, you can use the following information to open only those ports that are required for outbound communication by the Connector.



The source IP address is the Connector host.

Service	Port	Protocol	Destination	Purpose
Active Directory	88	TCP	Active Directory forest	Kerberos V authentication
	139	TCP	Active Directory forest	NetBIOS service session
	389	TCP	Active Directory forest	LDAP
	445	TCP	Active Directory forest	Microsoft SMB/CIFS over TCP with NetBIOS framing
	464	TCP	Active Directory forest	Kerberos V change & set password (SET_CHANGE)
	749	TCP	Active Directory forest	Active Directory Kerberos V change & set password (RPCSEC_GSS)
	137	UDP	Active Directory forest	NetBIOS name service
	138	UDP	Active Directory forest	NetBIOS datagram service
	464	UDP	Active Directory forest	Kerberos key administration
API calls and AutoSupport	443	HTTPS	Outbound internet and ONTAP cluster management LIF	API calls to AWS and ONTAP, and sending AutoSupport messages to NetApp
DNS	53	UDP	DNS	Used for DNS resolve by Cloud Manager

Rules for the Connector in GCP

The firewall rules for the Connector requires both inbound and outbound rules.

Inbound rules

Protocol	Port	Purpose
SSH	22	Provides SSH access to the Connector host
HTTP	80	Provides HTTP access from client web browsers to the local user interface
HTTPS	443	Provides HTTPS access from client web browsers to the local user interface

Outbound rules

The predefined firewall rules for the Connector opens all outbound traffic. If that is acceptable, follow the basic outbound rules. If you need more rigid rules, use the advanced outbound rules.

Basic outbound rules

The predefined firewall rules for the Connector includes the following outbound rules.

Protocol	Port	Purpose
All TCP	All	All outbound traffic
All UDP	All	All outbound traffic

Advanced outbound rules

If you need rigid rules for outbound traffic, you can use the following information to open only those ports that are required for outbound communication by the Connector.



The source IP address is the Connector host.

Service	Protocol	Port	Destination	Purpose
Active Directory	TCP	88	Active Directory forest	Kerberos V authentication
	TCP	139	Active Directory forest	NetBIOS service session
	TCP	389	Active Directory forest	LDAP
	TCP	445	Active Directory forest	Microsoft SMB/CIFS over TCP with NetBIOS framing
	TCP	464	Active Directory forest	Kerberos V change & set password (SET_CHANGE)
	TCP	749	Active Directory forest	Active Directory Kerberos V change & set password (RPCSEC_GSS)
	UDP	137	Active Directory forest	NetBIOS name service
	UDP	138	Active Directory forest	NetBIOS datagram service
	UDP	464	Active Directory forest	Kerberos key administration
API calls and AutoSupport	HTTPS	443	Outbound internet and ONTAP cluster management LIF	API calls to GCP and ONTAP, and sending AutoSupport messages to NetApp
DNS	UDP	53	DNS	Used for DNS resolve by Cloud Manager

Ports for the on-prem Connector

The Connector uses the following *inbound* ports when installed manually on an on-premises Linux host.

These inbound rules apply to both deployment models for the on-prem Connector: installed with internet access or without internet access.

Protocol	Port	Purpose
HTTP	80	Provides HTTP access from client web browsers to the local user interface
HTTPS	443	Provides HTTPS access from client web browsers to the local user interface

Creating a Connector in AWS from Cloud Manager

An Account Admin needs to deploy a *Connector* before you can use most Cloud Manager features. [Learn when a Connector is required](#). The Connector enables Cloud Manager to manage resources and processes within your public cloud environment.

This page describes how to create a Connector in AWS directly from Cloud Manager. [Learn about other ways to deploy a Connector](#).

These steps must be completed by a user who has the Account Admin role. A Workspace Admin can't create a

Connector.



When you create your first Cloud Volumes ONTAP working environment, Cloud Manager will prompt you to create a Connector if you don't have one yet.

Setting up AWS permissions to create a Connector

Before you can deploy a Connector from Cloud Manager, you need to ensure that your AWS account has the correct permissions.

Steps

1. Download the Connector IAM policy from the following location:

[NetApp Cloud Manager: AWS, Azure, and GCP Policies](#)



For IAM user permissions for Amazon FSx for ONTAP, see [Create an FSx for ONTAP working environment](#).

2. From the AWS IAM console, create your own policy by copying and pasting the text from the Connector IAM policy.
3. Attach the policy that you created in the previous step to the IAM user who will create the Connector from Cloud Manager.

Result

The AWS user now has the permissions required to create the Connector from Cloud Manager. You'll need to specify AWS access keys for this user when you're prompted by Cloud Manager.

Creating a Connector in AWS

Cloud Manager enables you to create a Connector in AWS directly from its user interface.

What you'll need

- An AWS access key and secret key for an IAM user who has the [required permissions](#) to create a Connector.
- A VPC, subnet, and keypair in your AWS region of choice.
- If you don't want Cloud Manager to automatically create an IAM role for the Connector, then you'll need to create your own [using this policy](#).

These permissions are for the Connector instance. It's a different set of permissions than what's provided in the first bullet above.

Steps

1. If you're creating your first Working Environment, click **Add Working Environment** and follow the prompts. Otherwise, click the **Connector** drop-down and select **Add Connector**.



2. Choose **Amazon Web Services** as your cloud provider and click **Continue**.

Remember that the Connector must have a network connection to the type of working environment that you're creating and the services that you're planning to enable.

[Learn more about networking requirements for the Connector.](#)

3. Follow the steps in the wizard to create the Connector:

- **Get Ready:** Review what you'll need.
- **AWS Credentials:** Specify the AWS access key and secret key that meet permissions requirements and then select your region.
- **Details:** Provide details about the Connector.
 - Enter a name for the instance.
 - Add custom tags (metadata) to the instance.
 - Choose whether you want Cloud Manager to create a new role that has the required permissions, or if you want to select an existing role that you set up with [the required permissions](#).
 - Choose whether you want to encrypt the Connector's EBS disks. You have the option to use the default encryption key or to use a custom key.
- **Network:** Specify a VPC, subnet, and key pair for the instance, choose whether to enable a public IP address, and optionally specify a proxy configuration.
- **Security Group:** Choose whether to create a new security group or whether to select an existing security group that allows inbound HTTP, HTTPS, and SSH access.



There's no incoming traffic to the Connector, unless you initiate it. HTTP and HTTPS provide access to the [local UI](#), which you'll use in rare circumstances. SSH is only needed if you need to connect to the host for troubleshooting.

- **Review:** Review your selections to verify that your set up is correct.

4. Click **Add**.

The instance should be ready in about 7 minutes. You should stay on the page until the process is complete.

After you finish

You need to associate a Connector with workspaces so Workspace Admins can use those Connectors to

create Cloud Volumes ONTAP systems. If you only have Account Admins, then associating the Connector with workspaces isn't required. Account Admins have the ability to access all workspaces in Cloud Manager by default. [Learn more](#).

Creating a Connector in Azure from Cloud Manager

An Account Admin needs to deploy a *Connector* before you can use most Cloud Manager features. The Connector enables Cloud Manager to manage resources and processes within your public cloud environment. [Learn when a Connector is required](#).

This page describes how to create a Connector in Azure directly from Cloud Manager. [Learn about other ways to deploy a Connector](#).

These steps must be completed by a user who has the Account Admin role. A Workspace Admin can't create a Connector.



When you create your first Cloud Volumes ONTAP working environment, Cloud Manager will prompt you to create a Connector if you don't have one yet.

Overview

To deploy a Connector, you need to provide Cloud Manager with a login that has the required permissions to create the Connector VM in Azure.

You have two options:

1. Sign in with your Microsoft account when prompted. This account must have specific Azure permissions. This is the default option.

[Follow the steps below to get started.](#)

2. Provide details about an Azure AD service principal. This service principal also requires specific permissions.

[Follow the steps below to get started.](#)

Note about Azure regions

The Connector should be deployed in the same Azure region as the Cloud Volumes ONTAP systems that it manages, or in the [Azure region pair](#) for the Cloud Volumes ONTAP systems. This requirement ensures that an Azure Private Link connection is used between Cloud Volumes ONTAP and its associated storage accounts. [Learn how Cloud Volumes ONTAP uses an Azure Private Link](#).

Create a Connector using your Azure account

The default way to create a Connector in Azure is by logging in with your Azure account when prompted. The login form is owned and hosted by Microsoft. Your credentials are not provided to NetApp.

Set up permissions for your Azure account

Before you can deploy a Connector from Cloud Manager, you need to ensure that your Azure account has the correct permissions.

Steps

1. Download the [Azure policy for the Connector](#).



Right-click the link and click **Save link as...** to download the file.

2. Modify the JSON file by adding your Azure subscription ID to the assignable scope.

Example

```
"AssignableScopes": [  
  "/subscriptions/d333af45-0d07-4154-943d-c25fbzzzzzzz"  
]
```

3. Use the JSON file to create a custom role in Azure.

The following steps describe how to create the role by using Bash in Azure Cloud Shell.

- a. Start [Azure Cloud Shell](#) and choose the Bash environment.
- b. Upload the JSON file.



- c. Enter the following Azure CLI command:

```
az role definition create --role-definition  
Policy_for_Setup_As_Service_Azure.json
```

You should now have a custom role called *Azure SetupAsService*.

4. Assign the role to the user who will deploy the Connector from Cloud Manager:
 - a. Open the **Subscriptions** service and select the user's subscription.
 - b. Click **Access control (IAM)**.

c. Click **Add > Add role assignment** and then add the permissions:

- Select the **Azure SetupAsService** role and click **Next**.



Azure SetupAsService is the default name provided in the [Connector deployment policy for Azure](#). If you chose a different name for the role, then select that name instead.

- Keep **User, group, or service principal** selected.
- Click **Select members**, choose your user account, and click **Select**.
- Click **Next**.
- Click **Review + assign**.

Result

The Azure user now has the permissions required to deploy the Connector from Cloud Manager.

Create the Connector by logging in with your Azure account

Cloud Manager enables you to create a Connector in Azure directly from its user interface.

What you'll need

- An Azure subscription.
- A VNet and subnet in your Azure region of choice.
- If you don't want Cloud Manager to automatically create an Azure role for the Connector, then you'll need to create your own [using this policy](#).

These permissions are for the Connector instance itself. It's a different set of permissions than what you previously set up to simply deploy the Connector.

Steps

1. If you're creating your first Working Environment, click **Add Working Environment** and follow the prompts. Otherwise, click the **Connector** drop-down and select **Add Connector**.



2. Choose **Microsoft Azure** as your cloud provider.

Remember that the Connector must have a network connection to the type of working environment that you're creating and the services that you're planning to enable.

[Learn more about networking requirements for the Connector.](#)

3. Follow the steps in the wizard to create the Connector:

- **Get Ready:** Review what you'll need and click **Next**.
- If you're prompted, log in to your Microsoft account, which should have the required permissions to create the virtual machine.

The form is owned and hosted by Microsoft. Your credentials are not provided to NetApp.



If you're already logged in to an Azure account, then Cloud Manager will automatically use that account. If you have multiple accounts, then you might need to log out first to ensure that you're using the right account.

- **VM Authentication:** Choose an Azure subscription, a location, a new resource group or an existing resource group, and then choose an authentication method.
- **Details:** Enter a name for the instance, specify tags, and choose whether you want Cloud Manager to create a new role that has the required permissions, or if you want to select an existing role that you set up with [the required permissions](#).

Note that you can choose the subscriptions associated with this role. Each subscription that you choose provides the Connector with permissions to deploy Cloud Volumes ONTAP in those subscriptions.

- **Network:** Choose a VNet and subnet, whether to enable a public IP address, and optionally specify a proxy configuration.
- **Security Group:** Choose whether to create a new security group or whether to select an existing security group that allows inbound HTTP, HTTPS, and SSH access.



There's no incoming traffic to the Connector, unless you initiate it. HTTP and HTTPS provide access to the [local UI](#), which you'll use in rare circumstances. SSH is only needed if you need to connect to the host for troubleshooting.

- **Review:** Review your selections to verify that your set up is correct.

4. Click **Add**.

The virtual machine should be ready in about 7 minutes. You should stay on the page until the process is complete.

After you finish

You need to associate a Connector with workspaces so Workspace Admins can use those Connectors to create Cloud Volumes ONTAP systems. If you only have Account Admins, then associating the Connector with workspaces isn't required. Account Admins have the ability to access all workspaces in Cloud Manager by default. [Learn more](#).

Create a Connector using a service principal

Rather than logging in with your Azure account, you also have the option to provide Cloud Manager with the credentials for an Azure service principal that has the required permissions.

Granting Azure permissions using a service principal

Grant the required permissions to deploy a Connector in Azure by creating and setting up a service principal in Azure Active Directory and by obtaining the Azure credentials that Cloud Manager needs.

Steps

1. [Create an Azure Active Directory application](#).
2. [Assign the application to a role](#).
3. [Add Windows Azure Service Management API permissions](#).
4. [Get the application ID and directory ID](#).
5. [Create a client secret](#).

Create an Azure Active Directory application

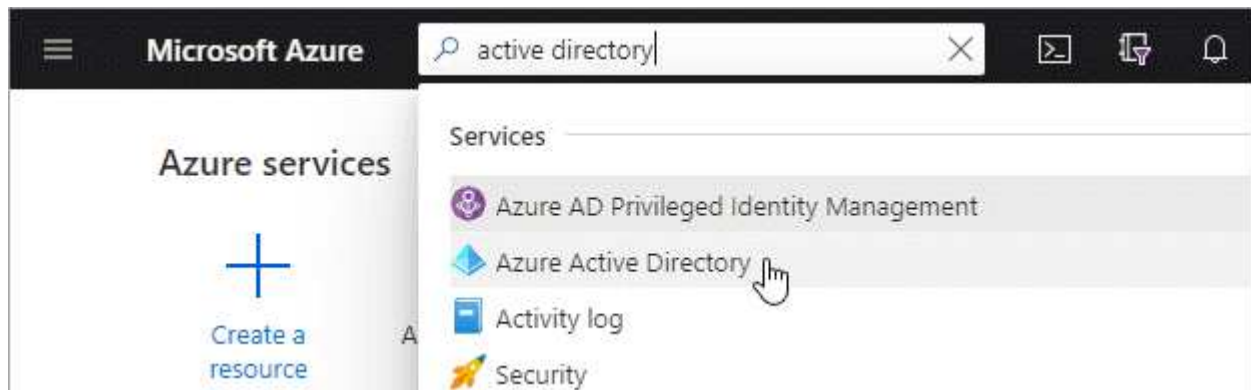
Create an Azure Active Directory (AD) application and service principal that Cloud Manager can use to deploy the Connector.

Before you begin

You must have the right permissions in Azure to create an Active Directory application and to assign the application to a role. For details, refer to [Microsoft Azure Documentation: Required permissions](#).

Steps

1. From the Azure portal, open the **Azure Active Directory** service.



2. In the menu, click **App registrations**.
3. Click **New registration**.
4. Specify details about the application:
 - **Name:** Enter a name for the application.
 - **Account type:** Select an account type (any will work with Cloud Manager).
 - **Redirect URI:** You can leave this field blank.
5. Click **Register**.

Result

You've created the AD application and service principal.

Assign the application to a role

You must bind the service principal to the Azure subscription in which you plan to deploy the Connector and assign it the custom "Azure SetupAsService" role.

Steps

1. Download the [Connector deployment policy for Azure](#).



Right-click the link and click **Save link as...** to download the file.

2. Modify the JSON file by adding your Azure subscription ID to the assignable scope.

Example

```
"AssignableScopes": [  
  "/subscriptions/398e471c-3b42-4ae7-9b59-ce5bbzzzzzzz"
```

3. Use the JSON file to create a custom role in Azure.

The following steps describe how to create the role by using Bash in Azure Cloud Shell.

- a. Start [Azure Cloud Shell](#) and choose the Bash environment.
- b. Upload the JSON file.



- c. Enter the following Azure CLI command:

```
az role definition create --role-definition  
Policy_for_Setup_As_Service_Azure.json
```

You should now have a custom role called *Azure SetupAsService*.

4. Assign the application to the role:
 - a. From the Azure portal, open the **Subscriptions** service.
 - b. Select the subscription.
 - c. Click **Access control (IAM) > Add > Add role assignment**.
 - d. In the **Role** tab, select the **Cloud Manager Operator** role and click **Next**.
 - e. In the **Members** tab, complete the following steps:
 - Keep **User, group, or service principal** selected.
 - Click **Select members**.



- Search for the name of the application.

Here's an example:



- Select the application and click **Select**.
 - Click **Next**.
- f. Click **Review + assign**.

The service principal now has the required Azure permissions to deploy the Connector.

Add Windows Azure Service Management API permissions

The service principal must have "Windows Azure Service Management API" permissions.

Steps

1. In the **Azure Active Directory** service, click **App registrations** and select the application.
2. Click **API permissions > Add a permission**.
3. Under **Microsoft APIs**, select **Azure Service Management**.

Request API permissions

Select an API

Microsoft APIs APIs my organization uses My APIs

Commonly used Microsoft APIs

Microsoft Graph
Take advantage of the tremendous amount of data in Office 365, Enterprise Mobility + Security, and Windows 10. Access Azure AD, Excel, Intune, Outlook/Exchange, OneDrive, OneNote, SharePoint, Planner, and more through a single endpoint.

 Azure Batch Schedule large-scale parallel and HPC applications in the cloud	 Azure Data Catalog Programmatic access to Data Catalog resources to register, annotate and search data assets	 Azure Data Explorer Perform ad-hoc queries on terabytes of data to build near real-time and complex analytics solutions
 Azure Data Lake Access to storage and compute for big data analytic scenarios	 Azure DevOps Integrate with Azure DevOps and Azure DevOps server	 Azure Import/Export Programmatic control of import/export jobs
 Azure Key Vault Manage your key vaults as well as the keys, secrets, and certificates within your Key Vaults	 Azure Rights Management Services Allow validated users to read and write protected content	 Azure Service Management Programmatic access to much of the functionality available through the Azure portal
 Azure Storage Secure, massively scalable object and data lake storage for unstructured and semi-structured data	 Customer Insights Create profile and interaction models for your products	 Data Export Service for Microsoft Dynamics 365 Export data from Microsoft Dynamics CRM organization to an external destination

4. Click **Access Azure Service Management as organization users** and then click **Add permissions**.

Request API permissions

[← All APIs](#)



Azure Service Management

<https://management.azure.com/> [Docs](#)

What type of permissions does your application require?

Delegated permissions

Your application needs to access the API as the signed-in user.

Application permissions

Your application runs as a background service or daemon without a signed-in user.

Select permissions

[expand all](#)

Type to search	
PERMISSION	ADMIN CONSENT REQUIRED
<input checked="" type="checkbox"/> user_impersonation Access Azure Service Management as organization users (preview) ⓘ	-

Get the application ID and directory ID

When you create the Connector from Cloud Manager, you need to provide the application (client) ID and the directory (tenant) ID for the application. Cloud Manager uses the IDs to programmatically sign in.

Steps

1. In the **Azure Active Directory** service, click **App registrations** and select the application.
2. Copy the **Application (client) ID** and the **Directory (tenant) ID**.



Create a client secret

You need to create a client secret and then provide Cloud Manager with the value of the secret so Cloud Manager can use it to authenticate with Azure AD.

Steps

1. Open the **Azure Active Directory** service.
2. Click **App registrations** and select your application.
3. Click **Certificates & secrets > New client secret**.

4. Provide a description of the secret and a duration.
5. Click **Add**.
6. Copy the value of the client secret.

Client secrets

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

+ New client secret		
DESCRIPTION	EXPIRES	VALUE
test secret	8/16/2020	*sZ1jSe2By:D*-ZRoV4NLfdAcY7:+0vA

Copy to clipboard

Result

Your service principal is now setup and you should have copied the application (client) ID, the directory (tenant) ID, and the value of the client secret. You need to enter this information in Cloud Manager when you create the Connector.

Create the Connector by logging in with the service principal

Cloud Manager enables you to create a Connector in Azure directly from its user interface.

What you'll need

- An Azure subscription.
- A VNet and subnet in your Azure region of choice.
- If you don't want Cloud Manager to automatically create an Azure role for the Connector, then you'll need to create your own [using this policy](#).

These permissions are for the Connector instance itself. It's a different set of permissions than what you previously set up to simply deploy the Connector.

Steps

1. If you're creating your first Working Environment, click **Add Working Environment** and follow the prompts. Otherwise, click the **Connector** drop-down and select **Add Connector**.



2. Choose **Microsoft Azure** as your cloud provider.

Remember that the Connector must have a network connection to the type of working environment that you're creating and the services that you're planning to enable.

[Learn more about networking requirements for the Connector.](#)

3. Follow the steps in the wizard to create the Connector:

- **Get Ready:** Click **Azure AD service principal** and enter information about the Azure Active Directory service principal that grants the required permissions:
- Application (client) ID: See [Get the application ID and directory ID](#).
- Directory (tenant) ID: See [Get the application ID and directory ID](#).
- Client Secret: See [Create a client secret](#).
- **VM Authentication:** Choose an Azure subscription, a location, a new resource group or an existing resource group, and then choose an authentication method.
- **Details:** Enter a name for the instance, specify tags, and choose whether you want Cloud Manager to create a new role that has the required permissions, or if you want to select an existing role that you set up with [the required permissions](#).

Note that you can choose the subscriptions associated with this role. Each subscription that you choose provides the Connector with permissions to deploy Cloud Volumes ONTAP in those subscriptions.

- **Network:** Choose a VNet and subnet, whether to enable a public IP address, and optionally specify a proxy configuration.
- **Security Group:** Choose whether to create a new security group or whether to select an existing security group that allows inbound HTTP, HTTPS, and SSH access.



There's no incoming traffic to the Connector, unless you initiate it. HTTP and HTTPS provide access to the [local UI](#), which you'll use in rare circumstances. SSH is only needed if you need to connect to the host for troubleshooting.

- **Review:** Review your selections to verify that your set up is correct.

4. Click **Add**.

The virtual machine should be ready in about 7 minutes. You should stay on the page until the process is complete.

After you finish

You need to associate a Connector with workspaces so Workspace Admins can use those Connectors to create Cloud Volumes ONTAP systems. If you only have Account Admins, then associating the Connector with workspaces isn't required. Account Admins have the ability to access all workspaces in Cloud Manager by default. [Learn more](#).

Creating a Connector in GCP from Cloud Manager

An Account Admin needs to deploy a *Connector* before you can use most Cloud Manager features. [Learn when a Connector is required](#). The Connector enables Cloud Manager to manage resources and processes within your public cloud environment.

This page describes how to create a Connector in GCP directly from Cloud Manager. [Learn about other ways](#)

to deploy a Connector.

These steps must be completed by a user who has the Account Admin role. A Workspace Admin can't create a Connector.



When you create your first Cloud Volumes ONTAP working environment, Cloud Manager will prompt you to create a Connector if you don't have one yet.

Setting up permissions

Before you can deploy a Connector, you need to ensure that your GCP account has the correct permissions and that a service account is set up for the Connector VM.

Steps

1. Ensure that the GCP user who deploys the Connector has the permissions in the [Connector deployment policy for GCP](#).

You can create a custom role using the [YAML file](#) and then attach it to the user. You'll need to use the `gcloud` command line to create the role.

2. Set up a service account that has the permissions that Cloud Manager needs to create and manage Cloud Volumes ONTAP systems in projects.

You'll associate this service account with the Connector VM when you create it.

- a. [Create a role in GCP](#) that includes the permissions defined in the [Cloud Manager policy for GCP](#). Again, you'll need to use the `gcloud` command line.

The permissions contained in this YAML file are different than the permissions in step 1.

- b. [Create a GCP service account and apply the custom role that you just created](#).
- c. If you want to deploy Cloud Volumes ONTAP in other projects, [grant access by adding the service account with the Cloud Manager role to that project](#). You'll need to repeat this step for each project.

Result

The GCP user now has the permissions required to create the Connector and the service account for the Connector VM is set up.

Shared VPC permissions

If you are using a shared VPC to deploy resources into a service project, then the following permissions are required. This table is for reference and your environment should reflect the permissions table when IAM configuration is complete.

Service Account	Creator	Hosted in	Service project permissions	Host project permissions	Purpose
Cloud Manager service account	Custom	Service project	<ul style="list-style-type: none"> • The permissions found in this .yaml file 	<ul style="list-style-type: none"> • compute.networkUser • deploymentmanager.editor 	Deploying and maintaining Cloud Volumes ONTAP and services in the service project
Cloud Volumes ONTAP service account	Custom	Service project	<ul style="list-style-type: none"> • storage.admin • member: Cloud Manager service account as serviceAccount.user 	N/A	(Optional) For data tiering and Cloud Backup
Google APIs service agent	GCP	Service project	<ul style="list-style-type: none"> • (Default) Editor 	<ul style="list-style-type: none"> • compute.networkUser 	Interacts with GCP APIs on behalf of deployment. Allows Cloud Manager to use the shared network.
Google Compute Engine default service account	GCP	Service project	<ul style="list-style-type: none"> • (Default) Editor 	<ul style="list-style-type: none"> • compute.networkUser 	Deploys GCP instances and compute infrastructure on behalf of deployment. Allows Cloud Manager to use the shared network.

Notes:

1. deploymentmanager.editor is only required at the host project if you are not passing firewall rules to the deployment and are choosing to let Cloud Manager create them for you.
2. firewall.create and firewall.delete are only required if you are not passing firewall rules to the deployment and are choosing to let Cloud Manager create them for you.
3. For data tiering, the tiering service account must have the serviceAccount.user role on the service account, not just at the project level. Currently if you assign serviceAccount.user at the project level, the permissions don't show when you query the service account with getIAMPolicy.

Enabling Google Cloud APIs

Several APIs are required to deploy the Connector and Cloud Volumes ONTAP.

Step

1. [Enable the following Google Cloud APIs in your project.](#)
 - Cloud Deployment Manager V2 API
 - Cloud Logging API
 - Cloud Resource Manager API
 - Compute Engine API

- Identity and Access Management (IAM) API

Creating a Connector in GCP

Create a Connector in Google Cloud directly from the Cloud Manager user interface or by using `gcloud`.

What you'll need

- The [required permissions](#) for your Google Cloud account, as described in the first section of this page.
- A Google Cloud project.
- A service account that has the required permissions to create and manage Cloud Volumes ONTAP, as described in the first section of this page.
- A VPC and subnet in your Google Cloud region of choice.

Cloud Manager

1. If you're creating your first Working Environment, click **Add Working Environment** and follow the prompts. Otherwise, click the **Connector** drop-down and select **Add Connector**.



2. Choose **Google Cloud Platform** as your cloud provider.

Remember that the Connector must have a network connection to the type of working environment that you're creating and the services that you're planning to enable.

[Learn more about networking requirements for the Connector.](#)

3. Follow the steps in the wizard to create the Connector:

- **Get Ready:** Review what you'll need.
- If you're prompted, log in to your Google account, which should have the required permissions to create the virtual machine instance.

The form is owned and hosted by Google. Your credentials are not provided to NetApp.

- **Basic Settings:** Enter a name for the virtual machine instance, specify tags, select a project, and then select the service account that has the required permissions (refer to the section above for details).
- **Location:** Specify a region, zone, VPC, and subnet for the instance.
- **Network:** Choose whether to enable a public IP address and optionally specify a proxy configuration.
- **Firewall Policy:** Choose whether to create a new firewall policy or whether to select an existing firewall policy that allows inbound HTTP, HTTPS, and SSH access.



There's no incoming traffic to the Connector, unless you initiate it. HTTP and HTTPS provide access to the [local UI](#), which you'll use in rare circumstances. SSH is only needed if you need to connect to the host for troubleshooting.

- **Review:** Review your selections to verify that your set up is correct.

4. Click **Add**.

The instance should be ready in about 7 minutes. You should stay on the page until the process is complete.

gcloud

1. Log in to the gcloud SDK using your preferred methodology.

In our examples, we'll use a local shell with the gcloud SDK installed, but you could use the native Google Cloud Shell in the GCP console.

For more information about the Google Cloud SDK, visit the [Google Cloud SDK documentation page](#).

2. Verify that you are logged in as a user who has the required permissions that are defined in the section above:

```
gcloud auth list
```

The output should show the following where the * user account is the desired user account to be logged in as:

```
Credentialed Accounts
ACTIVE  ACCOUNT
      some_user_account@domain.com
*      desired_user_account@domain.com
To set the active account, run:
$ gcloud config set account `ACCOUNT`
Updates are available for some Cloud SDK components. To install
them,
please run:
$ gcloud components update
```

3. Run the gcloud compute instances create command:

```
gcloud compute instances create <instance-name>
  --machine-type=n1-standard-4
  --image-project=netapp-cloudmanager
  --image-family=cloudmanager
  --scopes=cloud-platform
  --project=<project>
  --service-account=<<service-account>>
  --zone=<zone>
  --no-address
  --tags <network-tag>
  --network <network-path>
  --subnet <subnet-path>
  --boot-disk-kms-key <kms-key-path>
```

instance-name

The desired instance name for the VM instance.

project

(Optional) The project where you want to deploy the VM.

service-account

The service account specified in the output from step 2.

zone

The zone where you want to deploy the VM

no-address

(Optional) No external IP address is used (you need a cloud NAT or proxy to route traffic to the public internet)

network-tag

(Optional) Add network tagging to link a firewall rule using tags to the Connector instance

network-path

(Optional) Add the name of the network to deploy the Connector into (for a Shared VPC, you need the full path)

subnet-path

(Optional) Add the name of the subnet to deploy the Connector into (for a Shared VPC, you need the full path)

kms-key-path

(Optional) Add a KMS key to encrypt the Connector's disks (IAM permissions also need to be applied)

For more information about these flags, visit the [Google Cloud compute SDK documentation](#).

Running the command deploys the Connector using the NetApp golden image. The Connector instance and software should be running in approximately five minutes.

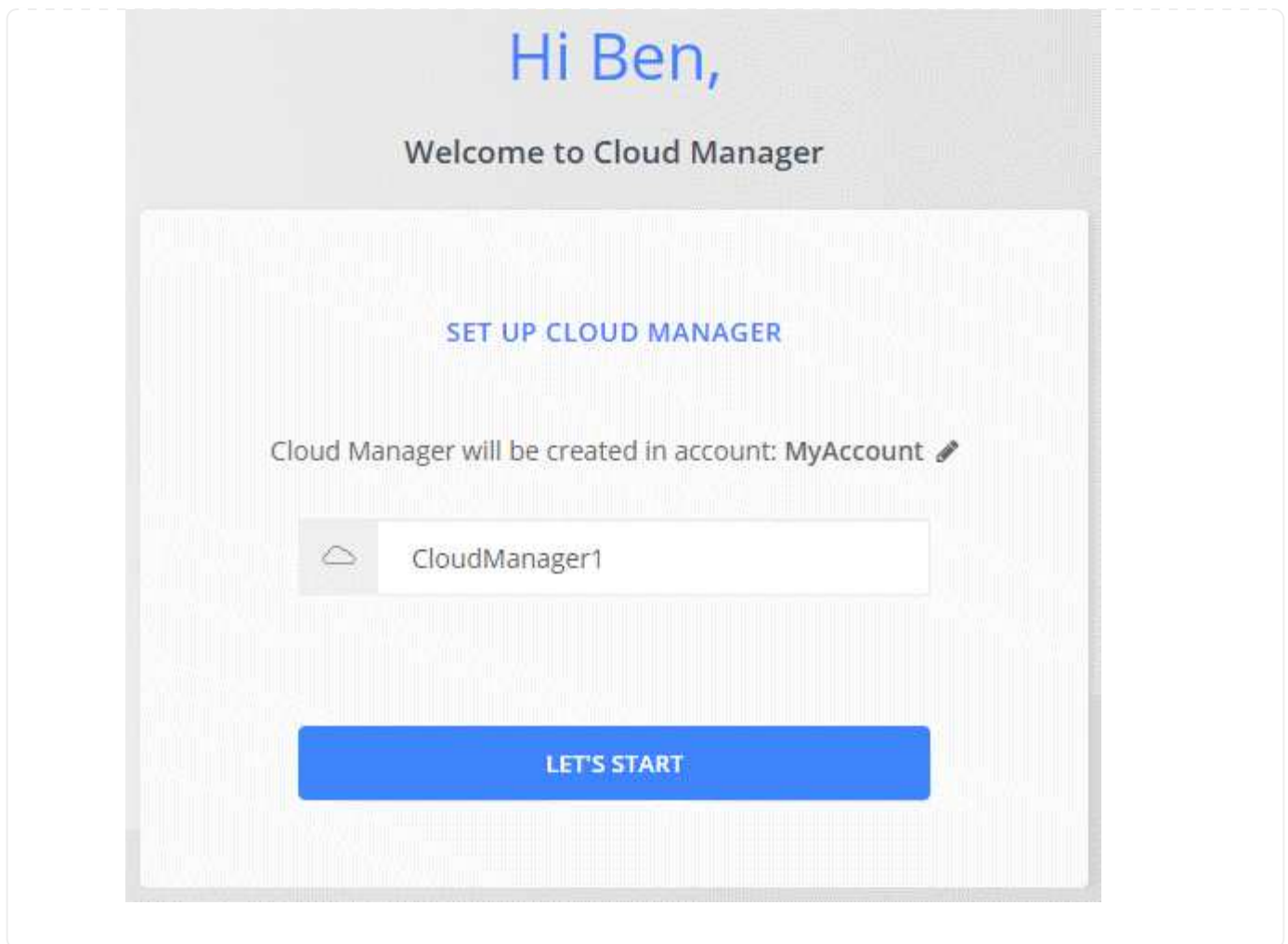
4. Open a web browser from a host that has a connection to the Connector instance and enter the following URL:

`http://ipaddress:80`

5. After you log in, set up the Connector:
 - a. Specify the NetApp account to associate with the Connector.

[Learn about NetApp accounts](#).

- b. Enter a name for the system.



Result

The Connector is now installed and set up with your NetApp account. Cloud Manager will automatically use this Connector when you create new working environments. But if you have more than one Connector, you'll need to [switch between them](#).

Install on your own host

Install the Connector on an existing Linux host that has internet access

The most common way to create a Connector is directly from Cloud Manager or from a cloud provider's marketplace. But you have the option to download and install the Connector software on an existing Linux host in your network or in the cloud. These steps are specific to hosts that have internet access.

[Learn about other ways to deploy a Connector.](#)



If you want to create a Cloud Volumes ONTAP system in Google Cloud, then you must have a Connector that's running in Google Cloud as well. You can't use a Connector that's running in AWS, Azure, or on-prem.

Verify host requirements

The Connector software must run on a host that meets specific operating system requirements, RAM requirements, port requirements, and so on.

A dedicated host is required

The Connector is not supported on a host that is shared with other applications. The host must be a dedicated host.

CPU

4 cores or 4 vCPUs

RAM

16 GB

AWS EC2 instance type

An instance type that meets the CPU and RAM requirements above. We recommend t3.xlarge and use that instance type when you deploy the Connector directly from Cloud Manager.

Azure VM size

An instance type that meets the CPU and RAM requirements above. We recommend DS3 v2 and use that VM size when you deploy the Connector directly from Cloud Manager.

GCP machine type

An instance type that meets the CPU and RAM requirements above. We recommend n1-standard-4 and use that machine type when you deploy the Connector directly from Cloud Manager.

Supported operating systems

- CentOS 7.6
- CentOS 7.7
- CentOS 7.8
- CentOS 7.9
- Red Hat Enterprise Linux 7.6
- Red Hat Enterprise Linux 7.7
- Red Hat Enterprise Linux 7.8
- Red Hat Enterprise Linux 7.9

The Red Hat Enterprise Linux system must be registered with Red Hat Subscription Management. If it is not registered, the system cannot access repositories to update required 3rd party software during Connector installation.

The Connector is supported on English-language versions of these operating systems.

Hypervisor

A bare metal or hosted hypervisor that is certified to run CentOS or Red Hat Enterprise Linux
[Red Hat Solution: Which hypervisors are certified to run Red Hat Enterprise Linux?](#)

Disk space in /opt

100 GiB of space must be available

Disk space in /var

20 GiB of space must be available

Outbound internet access

Outbound internet access is required to install the Connector and for the Connector to manage resources and processes within your public cloud environment. For a list of endpoints, see [Networking requirements for the Connector](#).

Install the Connector

After you verify that you have a supported Linux host, you can obtain the Connector software and then install it.

Required privileges

Root privileges are required to install the Connector.

About this task

- The installation installs the AWS command line tools (awscli) to enable recovery procedures from NetApp support.

If you receive a message that installing the awscli failed, you can safely ignore the message. The Connector can operate successfully without the tools.

- The installer that is available on the NetApp Support Site might be an earlier version. After installation, the Connector automatically updates itself if a new version is available.

Steps

1. Download the Cloud Manager software from the [NetApp Support Site](#), and then copy it to the Linux host.

For help with connecting and copying the file to an EC2 instance in AWS, see [AWS Documentation: Connecting to Your Linux Instance Using SSH](#).

2. Assign permissions to run the script.

```
chmod +x OnCommandCloudManager-V3.9.14.sh
```

3. Run the installation script.

If you have a proxy server, you will need to enter the command parameters as shown below. The installer doesn't prompt you to provide information about a proxy.

```
./OnCommandCloudManager-V3.9.14.sh [silent] [proxy=ipaddress]  
[proxyport=port] [proxyuser=user_name] [proxypwd=password]
```

silent runs the installation without prompting you for information.

proxy is required if the host is behind a proxy server.

proxyport is the port for the proxy server.

proxyuser is the user name for the proxy server, if basic authentication is required.

proxypwd is the password for the user name that you specified.

4. Unless you specified the silent parameter, enter **Y** to continue with the installation.

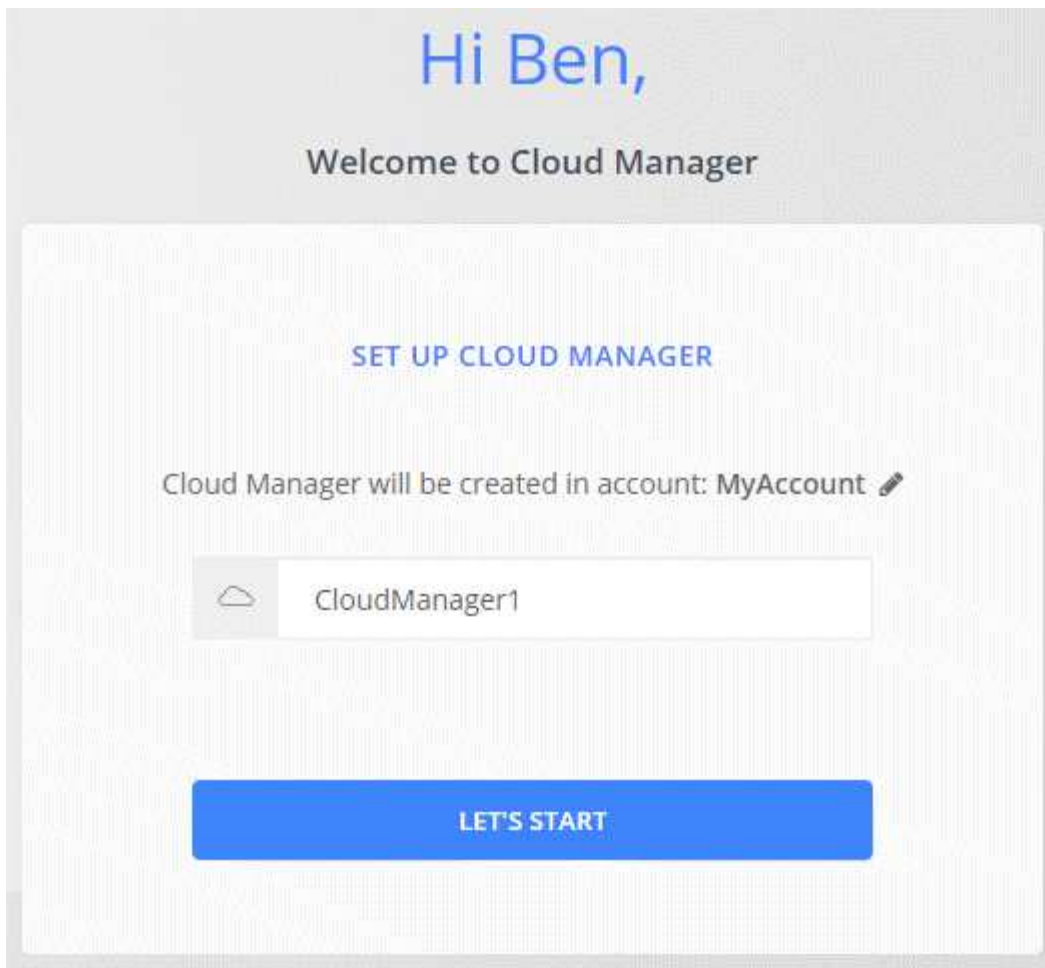
Cloud Manager is now installed. At the end of the installation, the Cloud Manager service (occm) restarts twice if you specified a proxy server.

5. Open a web browser and enter the following URL:

`https://ipaddress`

ipaddress can be localhost, a private IP address, or a public IP address, depending on the configuration of the host. For example, if the Connector is in the public cloud without a public IP address, you must enter a private IP address from a host that has a connection to the Connector host.

6. Sign up at NetApp Cloud Central or log in.
7. If you installed the Connector in Google Cloud, set up a service account that has the permissions that Cloud Manager needs to create and manage Cloud Volumes ONTAP systems in projects.
 - a. [Create a role in GCP](#) that includes the permissions defined in the [Cloud Manager policy for GCP](#).
 - b. [Create a GCP service account and apply the custom role that you just created](#).
 - c. [Associate this service account with the Connector VM](#).
 - d. If you want to deploy Cloud Volumes ONTAP in other projects, [grant access by adding the service account with the Cloud Manager role to that project](#). You'll need to repeat this step for each project.
8. After you log in, set up Cloud Manager:
 - a. Specify the NetApp account to associate with the Connector.
[Learn about NetApp accounts](#).
 - b. Enter a name for the system.



Result

The Connector is now installed and set up with your NetApp account. Cloud Manager will automatically use this Connector when you create new working environments.

After you finish

Set up permissions so Cloud Manager can manage resources and processes within your public cloud environment:

- AWS: [Set up an AWS account and then add it to Cloud Manager.](#)
- Azure: [Set up an Azure account and then add it to Cloud Manager.](#)
- Google Cloud: See step 7 above.

Install the Connector on-prem without internet access

You can install the Connector on an on-premises Linux host that doesn't have internet access. You can then discover on-prem ONTAP clusters, replicate data between them, and scan them with Cloud Data Sense.

These installation instructions are specifically for the use case described above. [Learn about other ways to deploy a Connector.](#)

Verify host requirements

The Connector software must run on a host that meets specific operating system requirements, RAM requirements, port requirements, and so on.

A dedicated host is required

The Connector is not supported on a host that is shared with other applications. The host must be a dedicated host.

CPU

4 cores or 4 vCPUs

RAM

16 GB

Supported operating systems

- CentOS 7.6
- CentOS 7.7
- CentOS 7.8
- CentOS 7.9
- Red Hat Enterprise Linux 7.6
- Red Hat Enterprise Linux 7.7
- Red Hat Enterprise Linux 7.8
- Red Hat Enterprise Linux 7.9

The Red Hat Enterprise Linux system must be registered with Red Hat Subscription Management. If it is not registered, the system cannot access repositories to update required 3rd party software during Connector installation.

The Connector is supported on English-language versions of these operating systems.

Hypervisor

A bare metal or hosted hypervisor that is certified to run CentOS or Red Hat Enterprise Linux
[Red Hat Solution: Which hypervisors are certified to run Red Hat Enterprise Linux?](#)

Disk type

An SSD is required

Disk space in /opt

100 GiB of space must be available

Disk space in /var

20 GiB of space must be available

Docker Engine

Docker Engine version 19 or later is required on the host before you install the Connector. [View installation instructions.](#)

Install the Connector

After you verify that you have a supported Linux host, you can obtain the Connector software and then install it.

Required privileges

Root privileges are required to install the Connector.

Steps

1. Verify that docker is enabled and running.

```
sudo systemctl enable docker && sudo systemctl start docker
```

2. Download the Cloud Manager software from the [NetApp Support Site](#).
3. Copy the installer to the Linux host.
4. Assign permissions to run the script.

```
chmod +x /path/cloud-manager-connector-offline-v3.9.14
```

5. Run the installation script:

```
sudo /path/cloud-manager-connector-offline-v3.9.14
```

6. Open a web browser and enter `https://ipaddress` where *ipaddress* is the IP address of the Linux host.

You should see the following screen.



7. Click **Set Up New Cloud Manager** and follow the prompts to set up the system.
 - **System Details:** Enter a name for the Cloud Manager system and your company name.

- **Create Admin User:** Create the admin user for the system.

This user account runs locally on the system. There's no connection to NetApp Cloud Central.

- **Review:** Review the details, accept the license agreement, and then click **Set Up**.

8. Log in to Cloud Manager using the admin user that you just created.

Result

The Connector is now installed and you can start using the Cloud Manager features that are available in a dark site deployment.

What's next?

- [Discover on-prem ONTAP clusters](#)
- [Replicate data between on-prem ONTAP clusters](#)
- [Scan volume data using Cloud Data Sense](#)

When new versions of the Connector software are available, they'll be posted to the NetApp Support Site. [Learn how to upgrade the Connector.](#)

Create from the marketplace

Creating a Connector from the AWS Marketplace

It's best to create a Connector directly from Cloud Manager, but you can launch a Connector from the AWS Marketplace, if you'd rather not specify AWS access keys. After you create and set up the Connector, Cloud Manager will automatically use it when you create new working environments.

Steps

1. Create an IAM policy and role for the EC2 instance:
 - a. Download the Cloud Manager IAM policy from the following location:

[NetApp Cloud Manager: AWS, Azure, and GCP Policies](#)

- b. From the IAM console, create your own policy by copying and pasting the text from the Cloud Manager IAM policy.
 - c. Create an IAM role with the role type Amazon EC2 and attach the policy that you created in the previous step to the role.
2. Now go to the [Cloud Manager page on the AWS Marketplace](#) to deploy Cloud Manager from an AMI.

The IAM user must have AWS Marketplace permissions to subscribe and unsubscribe.
3. On the Marketplace page, click **Continue to Subscribe** and then click **Continue to Configuration**.

a

Cloud Manager - Manual Installation without access keys

By: [NetApp, Inc.](#) Latest Version: 3.8.4

Read below for instructions on how to deploy Cloud Volumes ONTAP.

Linux/Unix ★★★★★ 6 AWS reviews

Typical Total Price: **\$0.226/hr**

Total pricing per instance for services hosted on t3.xlarge in US East (N. Virginia). [View Details](#)

Continue to Subscribe

Save to List

Overview Pricing Usage Support Reviews

Product Overview

Do NOT subscribe on this page unless instructed by NetApp or redirected here from the NetApp website.

This listing lets you manually launch a Cloud Manager instance without providing your AWS credentials. After launching the Cloud Manager software in AWS, you can access it by entering the instance's IP address in a web browser. If you subscribe here, you still need to subscribe on the listing below for PAYGO charges.

Highlights

- See Product Overview for instructions on how to deploy NetApp Cloud Manager.

b

Cloud Manager - Manual Installation without access keys

Continue to Configuration

[< Product Detail](#) [Subscribe](#)

Subscribe to this software

You're subscribed to this software. Please see the terms and pricing details below or click the button above to configure your software.

Terms and Conditions

NetApp, Inc. Offer

You have subscribed to this software and agreed that your use of this software is subject to the pricing terms and the seller's [End User License Agreement \(EULA\)](#). You agreed that AWS may share information about this transaction (including your payment terms) with the respective seller, reseller or underlying provider, as applicable, in accordance with the [AWS Privacy Notice](#). Your use of AWS services remains subject to the [AWS Customer Agreement](#) or other agreement with AWS governing your use of such services.

4. Change any of the default options and click **Continue to Launch**.

- Under **Choose Action**, select **Launch through EC2** and then click **Launch**.

These steps describe how to launch the instance from the EC2 Console because the console enables you to attach an IAM role to the Cloud Manager instance. This isn't possible using the **Launch from Website** action.

- Follow the prompts to configure and deploy the instance:

- **Choose Instance Type:** Depending on region availability, choose one of the supported instance types (t3.xlarge is recommended).

[Review the instance requirements.](#)

- **Configure Instance:** Select a VPC and subnet, choose the IAM role that you created in step 1, enable termination protection (recommended), and choose any other configuration options that meet your requirements.

The screenshot shows the AWS EC2 console configuration page for launching an instance. The page is divided into several sections, each with a title and a list of options. The 'IAM role' section is highlighted with a red box, showing the 'Cloud_Manager' role selected. The 'Enable termination protection' section is also highlighted with a red box, showing the 'Protect against accidental termination' checkbox checked. Other sections include 'Number of instances' (1), 'Purchasing option' (Request Spot instances), 'Network' (vpc-a76d91c2 | VPC4QA (default)), 'Subnet' (subnet-39536c13 | QASubnet1 | us-east-1b), 'Auto-assign Public IP' (Enable), 'Placement group' (Add instance to placement group), 'Capacity Reservation' (Open), 'CPU options' (Specify CPU options), 'Shutdown behavior' (Stop), and 'Monitoring' (Enable CloudWatch detailed monitoring).

- **Add Storage:** Keep the default storage options.
- **Add Tags:** Enter tags for the instance, if desired.
- **Configure Security Group:** Specify the required connection methods for the Connector instance: SSH, HTTP, and HTTPS.
- **Review:** Review your selections and click **Launch**.

AWS launches the software with the specified settings. The Connector instance and software should be running in approximately five minutes.

- Open a web browser from a host that has a connection to the Connector instance and enter the following URL:

`http://ipaddress:80`

8. After you log in, set up the Connector:
 - a. Specify the NetApp account to associate with the Connector.
[Learn about NetApp accounts.](#)
 - b. Enter a name for the system.



Result

The Connector is now installed and set up with your NetApp account. Cloud Manager will automatically use this Connector when you create new working environments. But if you have more than one Connector, you'll need to [switch between them](#).

Creating a Connector from the Azure Marketplace

It's best to create a Connector directly from Cloud Manager, but you can launch a Connector from the Azure Marketplace, if you prefer. After you create and set up the Connector, Cloud Manager will automatically use it when you create new working environments.

Creating a Connector in Azure

Deploy the Connector in Azure using the image in the Azure Marketplace and then log in to the Connector to specify your NetApp account.

Steps

1. [Go to the Azure Marketplace page for Cloud Manager.](#)
2. Click **Get it now** and then click **Continue**.
3. From the Azure portal, click **Create** and follow the steps to configure the virtual machine.

Note the following as you configure the VM:

- Cloud Manager can perform optimally with either HDD or SSD disks.
- Choose a VM size that meets CPU and RAM requirements. We recommend DS3 v2.

[Review the VM requirements.](#)

- For the network security group, the Connector requires inbound connections using SSH, HTTP, and HTTPS.

[Learn more about security group rules for the Connector.](#)

- Under **Management**, enable **System assigned managed identity** for the Connector by selecting **On**.

This setting is important because a managed identity allows the Connector virtual machine to identify itself to Azure Active Directory without providing any credentials. [Learn more about managed identities for Azure resources.](#)

4. On the **Review + create** page, review your selections and click **Create** to start the deployment.

Azure deploys the virtual machine with the specified settings. The virtual machine and Connector software should be running in approximately five minutes.

5. Open a web browser from a host that has a connection to the Connector virtual machine and enter the following URL:

`http://ipaddress:80`

6. After you log in, set up the Connector:
 - a. Specify the NetApp account to associate with the Connector.

[Learn about NetApp accounts.](#)

- b. Enter a name for the system.



Result

The Connector is now installed and set up. You must grant Azure permissions before users can deploy Cloud Volumes ONTAP in Azure.

Granting Azure permissions

When you deployed the Connector in Azure, you should have enabled a [system-assigned managed identity](#). You must now grant the required Azure permissions by creating a custom role and then by assigning the role to the Connector virtual machine for one or more subscriptions.

Steps

1. Create a custom role using the Cloud Manager policy:
 - a. Download the [Cloud Manager Azure policy](#).
 - b. Modify the JSON file by adding Azure subscription IDs to the assignable scope.

You should add the ID for each Azure subscription from which users will create Cloud Volumes ONTAP systems.

Example

```
"AssignableScopes": [  
  "/subscriptions/d333af45-0d07-4154-943d-c25fbzzzzzzz",  
  "/subscriptions/54b91999-b3e6-4599-908e-416e0zzzzzzz",  
  "/subscriptions/398e471c-3b42-4ae7-9b59-ce5bbzzzzzzz"
```

- c. Use the JSON file to create a custom role in Azure.

The following example shows how to create a custom role using the Azure CLI 2.0:

```
az role definition create --role-definition
C:\Policy_for_cloud_Manager_Azure_3.9.8.json
```

You should now have a custom role called Cloud Manager Operator that you can assign to the Connector virtual machine.

2. Assign the role to the Connector virtual machine for one or more subscriptions:

- a. Open the **Subscriptions** service and then select the subscription in which you want to deploy Cloud Volumes ONTAP systems.
- b. Click **Access control (IAM)** > **Add** > **Add role assignment**.
- c. In the **Role** tab, select the **Cloud Manager Operator** role and click **Next**.



Cloud Manager Operator is the default name provided in the [Cloud Manager policy](#). If you chose a different name for the role, then select that name instead.

- d. In the **Members** tab, complete the following steps:
 - Assign access to a **Managed identity**.
 - Click **Select members**, select the subscription in which the Connector virtual machine was created, choose **Virtual machine**, and then select the Connector virtual machine.
 - Click **Select**.
 - Click **Next**.
- e. Click **Review + assign**.
- f. If you want to deploy Cloud Volumes ONTAP from additional subscriptions, switch to that subscription and then repeat these steps.

Result

The Connector now has the permissions that it needs to manage resources and processes within your public cloud environment. Cloud Manager will automatically use this Connector when you create new working environments. But if you have more than one Connector, you'll need to [switch between them](#).

Copyright Information

Copyright © 2022 NetApp, Inc. All rights reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means-graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system-without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.277-7103 (October 1988) and FAR 52-227-19 (June 1987).

Trademark Information

NETAPP, the NETAPP logo, and the marks listed at <http://www.netapp.com/TM> are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.