



Administer and monitor

NetApp Console setup and administration

NetApp

January 27, 2026

This PDF was generated from <https://docs.netapp.com/us-en/console-setup-admin/task-adding-nss-accounts.html> on January 27, 2026. Always check docs.netapp.com for the latest.

Table of Contents

Administer and monitor	1
Associate NetApp Support accounts	1
Manage NSS credentials associated with NetApp Console	1
Manage credentials associated with your NetApp Console login	4
Console agents	5
Learn about NetApp Console agents	5
Deploy a Console agent	9
Maintain Console agents	155
Manage cloud provider credentials	168
Identity and access management	197
Learn about NetApp Console identity and access management	197
Get started with identity and access in NetApp Console	201
Set up your Console organization	202
Add users to your Console organization	211
Manage user access and security	215
NetApp Console access roles	220
Identity and access API	239
Security and compliance	241
Identity federation	241
Enforce ONTAP permissions for ONTAP Advanced View (ONTAP System Manager)	253
Enable read-only mode for a NetApp Console organization	253
Manage organization partnerships	255
Partnerships in NetApp Console	256
Manage partnerships in NetApp Console	259
Manage members for a partnership organization	261
Provide resource access to partnership users	262
Work in a partner organization	264
Monitor NetApp Console operations	264
Audit user activity from the Audit page	265
Monitor activities using the Notification Center	265

Administer and monitor

Associate NetApp Support accounts

Manage NSS credentials associated with NetApp Console

Associate a NetApp Support Site account with your Console organization to enable key workflows for storage management. These NSS credentials are associated with the entire organization.

The Console also supports associating one NSS account per user account. [Learn how to manage user-level credentials.](#)

Overview

Associating NetApp Support Site credentials with your specific Console account serial number is required to enable the following tasks:

- Deploying Cloud Volumes ONTAP when you bring your own license (BYOL)

Providing your NSS account is required so that the Console can upload your license key and to enable the subscription for the term that you purchased. This includes automatic updates for term renewals.

- Registering pay-as-you-go Cloud Volumes ONTAP systems

Providing your NSS account is required to activate support for your system and to gain access to NetApp technical support resources.

- Upgrading Cloud Volumes ONTAP software to the latest release

These credentials are associated with your specific Console account serial number. Users can access these credentials from **Support > NSS Management**.

Add an NSS account

You can add and manage your NetApp Support Site accounts for use with the Console from the Support Dashboard within the Console.

When you have added your NSS account, the Console uses this information for things like license downloads, software upgrade verification, and future support registrations.

You can associate multiple NSS accounts with your organization; however, you cannot have customer accounts and partner accounts within the same organization.



NetApp uses Microsoft Entra ID as the identity provider for authentication services specific to support and licensing.

Steps

1. In **Administration > Support**.
2. Select **NSS Management**.
3. Select **Add NSS Account**.

4. Select **Continue** to be redirected to a Microsoft login page.
5. At the login page, provide your NetApp Support Site registered email address and password.

Upon successful login, NetApp will store the NSS user name.

This is a system-generated ID that maps to your email. On the **NSS Management** page, you can display your email from the **...** menu.

- If you ever need to refresh your login credential tokens, there is also an **Update Credentials** option in the **...** menu.

Using this option prompts you to log in again. Note that the token for these accounts expire after 90 days. A notification will be posted to alert you of this.

What's next?

Users can now select the account when creating new Cloud Volumes ONTAP systems and when registering existing Cloud Volumes ONTAP systems.

- [Launching Cloud Volumes ONTAP in AWS](#)
- [Launching Cloud Volumes ONTAP in Azure](#)
- [Launching Cloud Volumes ONTAP in Google Cloud](#)
- [Registering pay-as-you-go systems](#)

Update NSS credentials

For security reasons, you must update your NSS credentials every 90 days. You'll be notified in the Console notification center if your NSS credential has expired. [Learn about the Notification Center](#).

Expired credentials can disrupt the following, but are not limited to:

- License updates, which mean you won't be able to take advantage of newly purchased capacity.
- Ability to submit and track support cases.

Additionally, you can update the NSS credentials associated with your organization if you want to change the NSS account associated with your organization. For example, if the person associated with your NSS account has left your company.

Steps

1. In **Administration > Support**.
2. Select **NSS Management**.
3. For the NSS account that you want to update, select **...** and then select **Update Credentials**.
4. When you're prompted, select **Continue** to be redirected to a Microsoft login page.

NetApp uses Microsoft Entra ID as the identity provider for authentication services related to support and licensing.

5. At the login page, provide your NetApp Support Site registered email address and password.

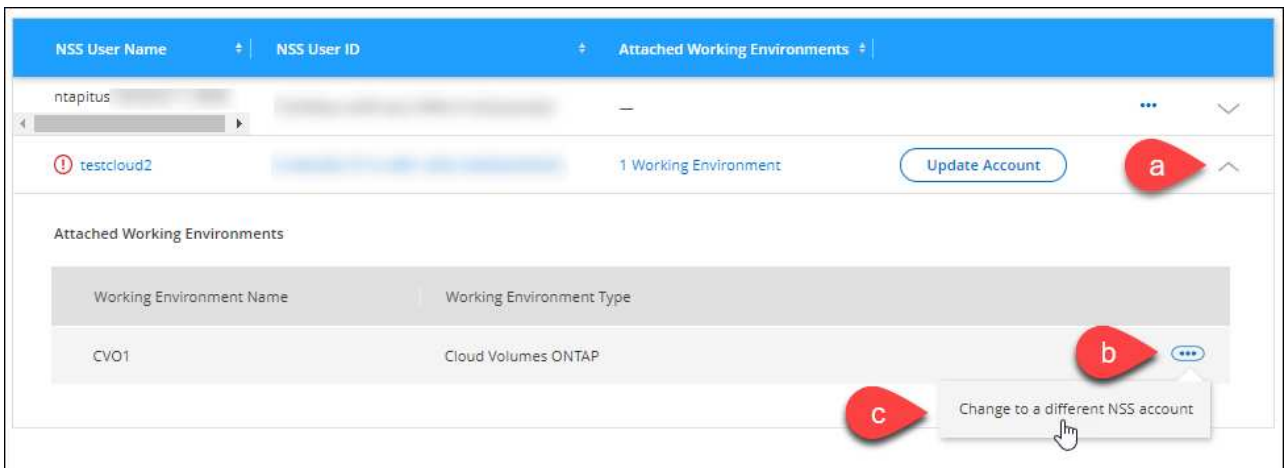
Attach a system to a different NSS account

If your organization has multiple NetApp Support Site accounts, you can change which account is associated with a Cloud Volumes ONTAP system.

You must first have associated the account with the Console.

Steps

1. In **Administration > Support**.
2. Select **NSS Management**.
3. Complete the following steps to change the NSS account:
 - a. Expand the row for the NetApp Support Site account that the system is currently associated with.
 - b. For the system that you want to change the association for, select **...**
 - c. Select **Change to a different NSS account**.



- d. Select the account and then select **Save**.

Display the email address for an NSS account

For security, the email address associated with an NSS account is not displayed by default. You can view the email address and associated user name for an NSS account.



When you go to the NSS Management page, the Console generates a token for each account in the table. That token includes information about the associated email address. The token is removed when you leave the page. The information is never cached, which helps protect your privacy.

Steps

1. In **Administration > Support**.
2. Select **NSS Management**.
3. For the NSS account that you want to update, select **...** and then select **Display Email Address**. You can use the copy button to copy the email address.

Remove an NSS account

Delete any of the NSS accounts that you no longer want to use with the Console.

You can't delete an account that is currently associated with a Cloud Volumes ONTAP system. You first need to [attach those systems to a different NSS account](#).

Steps

1. In **Administration > Support**.
2. Select **NSS Management**.
3. For the NSS account that you want to delete, select **...** and then select **Delete**.
4. Select **Delete** to confirm.

Manage credentials associated with your NetApp Console login

Depending on the actions that you've taken in the Console, you might have associated ONTAP credentials and NetApp Support Site (NSS) credentials with your user login. You can view and manage those credentials after you've associated them. For example, if you change the password for these credentials, then you'll need to update the password in the Console.

ONTAP credentials

Users need ONTAP admin credentials to discover ONTAP clusters in the Console. However, ONTAP System Manager access depends on whether or not you are using a Console agent.

Without a Console agent

Users are prompted to enter their ONTAP credentials to access ONTAP System Manager for the cluster. Users can choose to save these credentials in the Console which means they won't be prompted to enter them each time. User credentials are only visible to the respective user and can be managed from the User credentials page.

With a Console agent

By default, users are not prompted to enter their ONTAP credentials to access ONTAP System Manager. However, a Console administrator (with the Organization admin role) can configure the Console to prompt users to enter their ONTAP credentials. When this setting is enabled, users need enter their ONTAP credentials each time.

[Learn more.](#)

NSS credentials

The NSS credentials associated with your NetApp Console login enable support registration, case management, and access to Digital Advisor.

- When you access **Support > Resources** and register for support, you're prompted to associate NSS credentials with your login.

This registers your organization or account for support and activates support entitlement. Only one user in your organization must associate a NetApp Support Site account with their login to register for support and activate support entitlement. After this is completed, the **Resources** page shows that your account is registered for support.

[Learn how to register for support](#)

- When you access **Administration > Support > Case Management**, you're prompted to enter your NSS credentials, if you haven't already done so. This page enables you to create and manage the support cases associated with your NSS account and with your company.
- When you access Digital Advisor in the Console, you're prompted to log in to Digital Advisor by entering your NSS credentials.

Note the following about the NSS account associated with your login:

- The account is managed at the user level, which means it isn't viewable by other users who log in.
- There can be only one NSS account associated with Digital Advisor and support case management, per user.
- If you're trying to associate a NetApp Support Site account with a Cloud Volumes ONTAP system, you can only choose from the NSS accounts that were added to the organization that you are a member of.

NSS account-level credentials are different than the NSS account that's associated with your login. NSS account-level credentials enable you to deploy Cloud Volumes ONTAP with BYOL, register PAYGO systems, and upgrade its software.

[Learn more about using NSS credentials with your NetApp Console organization or account.](#)

Manage your user credentials

Manage your user credentials by updating the user name and password or by deleting the credentials.

Steps

1. Select **Administration > Credentials**.
2. Select **User Credentials**.
3. If you don't have any user credentials yet, you can select **Add NSS credentials** to add your NetApp Support Site account.
4. Manage existing credentials by choosing the following options from the Actions menu:
 - **Update credentials:** Update the user name and password for the account.
 - **Delete credentials:** Remove the NSS account associated with your Console login.

Console agents

Learn about NetApp Console agents

You use a Console agent to connect NetApp Console to your infrastructure and securely orchestrate storage solutions across AWS, Azure, Google Cloud, or on-premises environments, as well as use data protection services.

A Console agent enables you to:

- Orchestrate storage management tasks from the NetApp Console such as provisioning Cloud Volumes ONTAP, setting up storage volumes, using data classification, and more.
- Authenticate using your cloud provider's IAM roles for subscription billing integration

- Use advanced data services (NetApp Backup and Recovery, NetApp Disaster Recovery, NetApp Ransomware Resilience, and NetApp Cloud Tiering)
- Use the Console in restricted mode.

If you don't need advanced orchestration or data protection, you can centrally manage on-premises ONTAP clusters and cloud-native storage services without deploying an agent. Monitoring and data mobility tools are also available.

The following table shows which features and services you can use with and without a Console agent.

	Available with agent	Available without agent
Supported Storage systems:		
Amazon FSx for ONTAP	Yes (discovery and management features)	Yes (discovery only)
Amazon S3 storage	Yes	No
Azure Blob storage	Yes	Yes
Azure NetApp Files	Yes	Yes
Cloud Volumes ONTAP	Yes	No
E-Series systems	Yes	No
Google Cloud NetApp Volumes	Yes	Yes
Google Cloud storage buckets	Yes	No
StorageGRID systems	Yes	No
On-premises ONTAP cluster (advanced management and discovery)	Yes (advanced management and discovery)	No (basic discovery only)
Available storage management services:		
Alerts	Yes	No
Automation hub	Yes	Yes
Digital Advisor (Active IQ)	Yes	No
License and subscription management	Yes	No
Economic efficiency	Yes	No

	Available with agent	Available without agent
Home page dashboard metrics	Yes ²	No
Lifecycle planning	Yes	No ¹
Sustainability	Yes	No
Software updates	Yes	Yes
NetApp Workloads	Yes	Yes
Available data services:		
NetApp Backup and Recovery	Yes	No
Data Classification	Yes	No
NetApp Cloud Tiering	Yes	No
NetApp Copy and Sync	Yes	No
NetApp Disaster Recovery	Yes	No
NetApp Ransomware Resilience	Yes	No
NetApp Volume Caching	Yes	No

¹ You can view Lifecycle planning without a Console agent, but a Console agent is required to initiate actions.

² Accurate metrics on the Home page require appropriately sized and configured Console agents.

Console agents must be operational at all times

Console agents are a fundamental part of the NetApp Console. It's your responsibility (the customer) to ensure that relevant agents are up, operational, and accessible at all times. The Console can handle short agent outages, but you must fix infrastructure failures quickly.

This documentation is governed by the EULA. Operating the product outside the documentation may impact its functionality and your EULA rights.

Supported locations

You can install agents in the following locations:

- Amazon Web Services
- Microsoft Azure

Deploy a Console agent in Azure in the same region as the Cloud Volumes ONTAP systems it manages.

Alternatively, deploy it in the [Azure region pair](#). This 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](#)

- Google Cloud

To use the Console and data services with Google Cloud, deploy your agent in Google Cloud.

- On your premises

Communication with cloud providers

The agent uses TLS 1.3 for all communication to AWS, Azure, and Google Cloud.

Restricted mode

To use the Console in restricted mode, you install a Console agent and access the Console interface that's running locally on the Console agent.

[Learn about NetApp Console deployment modes.](#)

How to install a Console agent

You can install a Console agent directly from the Console, from your cloud provider's marketplace, or by manually installing the software on your own Linux host or in your VCenter environment.

- [Learn about NetApp Console deployment modes](#)
- [Get started with NetApp Console in standard mode](#)
- [Get started with NetApp Console in restricted mode](#)

Cloud provider permissions

You need specific permissions to create the Console agent directly from the NetApp Console and another set of permissions for the Console agent itself. If you create the Console agent in AWS or Azure directly from the Console, then the Console creates the Console agent with the permissions that it needs.

When using the Console in standard mode, how you provide permissions depends on how you plan to create the Console agent.

To learn how to set up permissions, refer to the following:

- Standard mode
 - [Agent installation options in AWS](#)
 - [Agent installation options in Azure](#)
 - [Agent installation options in Google Cloud](#)
 - [Set up cloud permissions for on-premises deployments](#)
- [Set up permissions for restricted mode](#)

To view the exact permissions that the Console agent needs for day-to-day operations, refer to the following pages:

- [Learn how the Console agent uses AWS permissions](#)

- [Learn how the Console agent uses Azure permissions](#)
- [Learn how the Console agent uses Google Cloud permissions](#)

It's your responsibility to update the Console agent policies as new permissions are added in subsequent releases. The release notes list new permissions.

Agent upgrades

NetApp updates agent software monthly to add features and improve stability. Some Console features, like Cloud Volumes ONTAP and on-premises ONTAP cluster management, rely on the Console agent version and settings.

When you install your agent in the cloud, the Console agent updates automatically if it has internet access.

Operating system and VM maintenance

Maintaining the operating system on the Console agent host is your (customer's) responsibility. For example, you (customer) should apply security updates to the operating system on the Console agent host by following your company's standard procedures for operating system distribution.

Note that you (customer) don't need to stop any services on the Console agent host when applying minor security updates.

If you (customer) need to stop and then start the Console agent VM, you should do so from your cloud provider's console or by using the standard procedures for on-premises management.

[The Console agent must be operational at all times.](#)

Multiple systems and agents

An agent can manage multiple systems and support data services in the Console. You can use a single agent to manage multiple systems based on deployment size and the data services you use.

For large-scale deployments, work with your NetApp representative to size your environment. Contact NetApp Support if you experience issues.

Here are a few examples of agent deployments:

- You have a multicloud environment (for example, AWS and Azure) and you prefer to have one agent in AWS and another in Azure. Each manages the Cloud Volumes ONTAP systems running in those environments.
- A service provider might use one Console organization to provide services for their customers, while using another organization to provide disaster recovery for one of their business units. Each organization needs its own agent.

Deploy a Console agent

AWS

Console agent installation options in AWS

There are a few different ways to create a Console agent in AWS. Directly from the NetApp Console is the most common way.

The following installation options are available:

- [Create the Console agent directly from the Console](#) (this is the standard option)

This action launches an EC2 instance running Linux and the Console agent software in a VPC of your choice.

- [Create a Console agent from the AWS Marketplace](#)

This action also launches an EC2 instance running Linux and the Console agent software, but the deployment is initiated directly from the AWS Marketplace, rather than from the Console.

- [Download and manually install the software on your own Linux host](#)

The installation option that you choose impacts how you prepare for installation. This includes how you provide the Console with the required permissions that it needs to authenticate and manage resources in AWS.

Create a Console agent in AWS from NetApp Console

You can create a Console agent in AWS directly from the NetApp Console. Before creating a Console agent in AWS from the Console, you need to set up your networking and prepare AWS permissions.

Before you begin

- You should have an [understanding of Console agents](#).
- You should review [Console agent limitations](#).

Step 1: Set up networking for deploying a Console agent in AWS

Ensure that the network location where you plan to install the Console agent supports the following requirements. These requirements enable the Console agent to manage resources and processes in your hybrid cloud.

VPC and subnet

When you create the Console agent, you need to specify the VPC and subnet where it should reside.

Connections to target networks

The Console agent requires a network connection to the location where you're planning to create and manage systems. For example, the network where you plan to create Cloud Volumes ONTAP systems or a storage system in your on-premises environment.

Outbound internet access

The network location where you deploy the Console agent must have an outbound internet connection to contact specific endpoints.

Endpoints contacted from the Console agent

The Console agent requires outbound internet access to contact the following endpoints to manage resources and processes within your public cloud environment for day-to-day operations.

The endpoints listed below are all CNAME entries.

Endpoints	Purpose
<p>AWS services (amazonaws.com):</p> <ul style="list-style-type: none"> • CloudFormation • Elastic Compute Cloud (EC2) • Identity and Access Management (IAM) • Key Management Service (KMS) • Security Token Service (STS) • Simple Storage Service (S3) 	<p>To manage AWS resources. The endpoint depends on your AWS region. Refer to AWS documentation for details</p>
<p>Amazon FsX for NetApp ONTAP:</p> <ul style="list-style-type: none"> • api.workloads.netapp.com 	<p>The web-based console contacts this endpoint to interact with the Workload Factory APIs to manage and operate FSx for ONTAP based workloads.</p>
https://mysupport.netapp.com	<p>To obtain licensing information and to send AutoSupport messages to NetApp support.</p>
https://signin.b2c.netapp.com	<p>To update NetApp Support Site (NSS) credentials or to add new NSS credentials to the NetApp Console.</p>
https://support.netapp.com	<p>To obtain licensing information and to send AutoSupport messages to NetApp support as well as to receive software updates for Cloud Volumes ONTAP.</p>
<p>https://api.blueexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.blueexp.netapp.com https://cdn.auth0.com</p>	<p>To provide features and services within the NetApp Console.</p>
<p>https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io</p>	<p>To obtain images for Console agent upgrades.</p> <ul style="list-style-type: none"> • When you deploy a new agent, the validation check tests connectivity to current endpoints. If you use previous endpoints, the validation check fails. To avoid this failure, skip the validation check. <p>Although the previous endpoints are still supported, NetApp recommends updating your firewall rules to the current endpoints as soon as possible. Learn how to update your endpoint list.</p> <ul style="list-style-type: none"> • When you update to the current endpoints in your firewall, your existing agents will continue to work.

Endpoints contacted from the NetApp console

As you use the web-based NetApp Console that's provided through the SaaS layer, it contacts several endpoints to complete data management tasks. This includes endpoints that are contacted to deploy the Console agent from the the Console.

[View the list of endpoints contacted from the NetApp console.](#)

Proxy server

NetApp supports both explicit and transparent proxy configurations. If you are using a transparent proxy, you only need to provide the certificate for the proxy server. If you are using an explicit proxy, you'll also need the IP address and credentials.

- IP address
- Credentials
- HTTPS certificate

Ports

There's no incoming traffic to the Console agent, unless you initiate it or if it is used as a proxy to send AutoSupport messages from Cloud Volumes ONTAP to NetApp Support.

- HTTP (80) and HTTPS (443) provide access to the local UI, which you'll use in rare circumstances.
- SSH (22) is only needed if you need to connect to the host for troubleshooting.
- Inbound connections over port 3128 are required if you deploy Cloud Volumes ONTAP systems in a subnet where an outbound internet connection isn't available.

If Cloud Volumes ONTAP systems don't have an outbound internet connection to send AutoSupport messages, the Console automatically configures those systems to use a proxy server that's included with the Console agent. The only requirement is to ensure that the Console agent's security group allows inbound connections over port 3128. You'll need to open this port after you deploy the Console agent.

Enable NTP

If you're planning to use NetApp Data Classification to scan your corporate data sources, you should enable a Network Time Protocol (NTP) service on both the Console agent and the NetApp Data Classification system so that the time is synchronized between the systems. [Learn more about NetApp Data classification](#)

You'll need to implement this networking requirement after you create the Console agent.

Step 2: Set up AWS permissions for the Console agent

The Console needs to authenticate with AWS before it can deploy the Console agent in your VPC. You can choose one of these authentication methods:

- Let the Console assume an IAM role that has the required permissions
- Provide an AWS access key and secret key for an IAM user who has the required permissions

With either option, the first step is to create an IAM policy. This policy contains only the permissions needed to launch the Console agent in AWS from the Console.

If needed, you can restrict the IAM policy by using the IAM `Condition` element. [AWS documentation: Condition element](#)

Steps

1. Go to the AWS IAM console.
2. Select **Policies > Create policy**.
3. Select **JSON**.
4. Copy and paste the following policy:

This policy contains only the permissions needed to launch the Console agent in AWS from the Console. When the Console creates the Console agent, it applies a new set of permissions to the Console agent that enables the Console agent to manage AWS resources. [View permissions required for the Console agent itself](#).

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iam:CreateRole",
        "iam:DeleteRole",
        "iam:PutRolePolicy",
        "iam:CreateInstanceProfile",
        "iam:DeleteRolePolicy",
        "iam:AddRoleToInstanceProfile",
        "iam:RemoveRoleFromInstanceProfile",
        "iam:DeleteInstanceProfile",
        "iam:PassRole",
        "iam:ListRoles",
        "ec2:DescribeInstanceStatus",
        "ec2:RunInstances",
        "ec2:ModifyInstanceAttribute",
        "ec2:CreateSecurityGroup",
        "ec2:DeleteSecurityGroup",
        "ec2:DescribeSecurityGroups",
        "ec2:RevokeSecurityGroupEgress",
        "ec2:AuthorizeSecurityGroupEgress",
        "ec2:AuthorizeSecurityGroupIngress",
        "ec2:RevokeSecurityGroupIngress",
        "ec2:CreateNetworkInterface",
        "ec2:DescribeNetworkInterfaces",
        "ec2:DeleteNetworkInterface",
        "ec2:ModifyNetworkInterfaceAttribute",
        "ec2:DescribeSubnets",
        "ec2:DescribeVpcs",
        "ec2:DescribeDhcpOptions",
        "ec2:DescribeKeyPairs",
        "ec2:DescribeRegions",
```

```

        "ec2:DescribeInstances",
        "ec2:CreateTags",
        "ec2:DescribeImages",
        "ec2:DescribeAvailabilityZones",
        "ec2:DescribeLaunchTemplates",
        "ec2:CreateLaunchTemplate",
        "cloudformation:CreateStack",
        "cloudformation>DeleteStack",
        "cloudformation:DescribeStacks",
        "cloudformation:DescribeStackEvents",
        "cloudformation:ValidateTemplate",
        "ec2:AssociateIamInstanceProfile",
        "ec2:DescribeIamInstanceProfileAssociations",
        "ec2:DisassociateIamInstanceProfile",
        "iam:GetRole",
        "iam:TagRole",
        "kms:ListAliases",
        "cloudformation:ListStacks"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "ec2:TerminateInstances"
    ],
    "Condition": {
        "StringLike": {
            "ec2:ResourceTag/OCCMInstance": "*"
        }
    },
    "Resource": [
        "arn:aws:ec2:*:*:instance/*"
    ]
}
]
}

```

5. Select **Next** and add tags, if needed.
6. Select **Next** and enter a name and description.
7. Select **Create policy**.
8. Either attach the policy to an IAM role that the Console can assume or to an IAM user so that you can provide the Console with access keys:
 - (Option 1) Set up an IAM role that the Console can assume:
 - a. Go to the AWS IAM console in the target account.

- b. Under Access Management, select **Roles > Create Role** and follow the steps to create the role.
- c. Under **Trusted entity type**, select **AWS account**.
- d. Select **Another AWS account** and enter the ID of the Console SaaS account: 952013314444
- e. Select the policy that you created in the previous section.
- f. After you create the role, copy the Role ARN so that you can paste it in the Console when you create the Console agent.
- (Option 2) Set up permissions for an IAM user so that you can provide the Console with access keys:
 - a. From the AWS IAM console, select **Users** and then select the user name.
 - b. Select **Add permissions > Attach existing policies directly**.
 - c. Select the policy that you created.
 - d. Select **Next** and then select **Add permissions**.
 - e. Ensure that you have the access key and secret key for the IAM user.

Result

You should now have an IAM role that has the required permissions or an IAM user that has the required permissions. When you create the Console agent from the Console, you can provide information about the role or access keys.

Step 3: Create the Console agent

Create the Console agent directly from the the Console web-based console.

About this task

- Creating the Console agent from the Console deploys an EC2 instance in AWS using a default configuration. Do not switch to a smaller EC2 instance with fewer CPUs or less RAM after creating the Console agent. [Learn about the default configuration for the Console agent](#).
- When the Console creates the Console agent, it creates an IAM role and a profile for the agent. This role includes permissions that enables the Console agent to manage AWS resources. Ensure the role is updated as new permissions are added in future releases.
[Learn more about the IAM policy for the Console agent](#).

Before you begin

You should have the following:

- An AWS authentication method: either an IAM role or access keys for an IAM user with the required permissions.
- A VPC and subnet that meets networking requirements.
- A key pair for the EC2 instance.
- Details about a proxy server, if a proxy is required for internet access from the Console agent.
- Set up [networking requirements](#).
- Set up [AWS permissions](#).

Steps

1. Select **Administration > Agents**.
2. On the **Overview** page, select **Deploy agent > AWS**

3. Follow the steps in the wizard to create the Console agent:
4. On the **Introduction** page provides an overview of the process
5. On the **AWS Credentials** page, specify your AWS region and then choose an authentication method, which is either an IAM role that the Console can assume or an AWS access key and secret key.



If you choose **Assume Role**, you can create the first set of credentials from the Console agent deployment wizard. Any additional set of credentials must be created from the Credentials page. They will then be available from the wizard in a drop-down list. [Learn how to add additional credentials.](#)

6. On the **Details** page, provide details about the Console agent.
 - Enter a name.
 - Add custom tags (metadata).
 - Choose whether you want the Console 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 Console agent's EBS disks. You have the option to use the default encryption key or to use a custom key.
7. On the **Network** page, Specify a VPC, subnet, and key pair for the agent, choose whether to enable a public IP address, and optionally specify a proxy configuration.

Ensure you have the correct key pair to access the Console agent virtual machine. Without a key pair, you cannot access it.

8. On the **Security Group** page, choose whether to create a new security group or whether to select an existing security group that allows the required inbound and outbound rules.

[View security group rules for AWS.](#)

9. Review your selections to verify that your set up is correct.
 - a. The **Validate agent configuration** check box is marked by default to have the Console validate the network connectivity requirements when you deploy. If the Console fails to deploy the agent, it provides a report to help you troubleshoot. If the deployment succeeds, no report is provided.

If you are still using the [previous endpoints](#) used for agent upgrades, the validation fails with an error. To avoid this, unmark the check box to skip the validation check.

10. Select **Add**.

The Console deploys the agent in about 10 minutes. Stay on the page until the process completes.

Result

After the process is complete, the Console agent is available for use from the Console.



If the deployment fails, you can download a report and logs from the Console to help you fix the issues. [Learn how to troubleshoot installation issues.](#)

If you have Amazon S3 buckets in the same AWS account where you created the Console agent, you'll see an Amazon S3 working environment appear on the **Systems** page automatically. [Learn how to manage S3](#)

Create a Console agent from the AWS Marketplace

You create a Console agent in AWS directly from the AWS Marketplace. To create a Console agent from the AWS Marketplace, you need to set up your networking, prepare AWS permissions, review instance requirements, and then create the Console agent.

Before you begin

- You should have an [understanding of Console agents](#).
- You should review [Console agent limitations](#).

Step 1: Set up networking

Ensure the network location for the Console agent meets the following requirements to manage hybrid cloud resources.

VPC and subnet

When you create the Console agent, you need to specify the VPC and subnet where it should reside.

Connections to target networks

The Console agent requires a network connection to the location where you're planning to create and manage systems. For example, the network where you plan to create Cloud Volumes ONTAP systems or a storage system in your on-premises environment.

Outbound internet access

The network location where you deploy the Console agent must have an outbound internet connection to contact specific endpoints.

Endpoints contacted from the Console agent

The Console agent requires outbound internet access to contact the following endpoints to manage resources and processes within your public cloud environment for day-to-day operations.

The endpoints listed below are all CNAME entries.

Endpoints	Purpose
AWS services (amazonaws.com): <ul style="list-style-type: none">• CloudFormation• Elastic Compute Cloud (EC2)• Identity and Access Management (IAM)• Key Management Service (KMS)• Security Token Service (STS)• Simple Storage Service (S3)	To manage AWS resources. The endpoint depends on your AWS region. Refer to AWS documentation for details
Amazon FsX for NetApp ONTAP: <ul style="list-style-type: none">• api.workloads.netapp.com	The web-based console contacts this endpoint to interact with the Workload Factory APIs to manage and operate FSx for ONTAP based workloads.

Endpoints	Purpose
https://mysupport.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support.
https://signin.b2c.netapp.com	To update NetApp Support Site (NSS) credentials or to add new NSS credentials to the NetApp Console.
https://support.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support as well as to receive software updates for Cloud Volumes ONTAP.
https://api.blueexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.blueexp.netapp.com https://cdn.auth0.com	To provide features and services within the NetApp Console.
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	<p>To obtain images for Console agent upgrades.</p> <ul style="list-style-type: none"> When you deploy a new agent, the validation check tests connectivity to current endpoints. If you use previous endpoints, the validation check fails. To avoid this failure, skip the validation check. <p>Although the previous endpoints are still supported, NetApp recommends updating your firewall rules to the current endpoints as soon as possible. Learn how to update your endpoint list.</p> <ul style="list-style-type: none"> When you update to the current endpoints in your firewall, your existing agents will continue to work.

Proxy server

NetApp supports both explicit and transparent proxy configurations. If you are using a transparent proxy, you only need to provide the certificate for the proxy server. If you are using an explicit proxy, you'll also need the IP address and credentials.

- IP address
- Credentials
- HTTPS certificate

Ports

There's no incoming traffic to the Console agent, unless you initiate it or if it is used as a proxy to send AutoSupport messages from Cloud Volumes ONTAP to NetApp Support.

- HTTP (80) and HTTPS (443) provide access to the local UI, which you'll use in rare circumstances.

- SSH (22) is only needed if you need to connect to the host for troubleshooting.
- Inbound connections over port 3128 are required if you deploy Cloud Volumes ONTAP systems in a subnet where an outbound internet connection isn't available.

If Cloud Volumes ONTAP systems don't have an outbound internet connection to send AutoSupport messages, the Console automatically configures those systems to use a proxy server that's included with the Console agent. The only requirement is to ensure that the Console agent's security group allows inbound connections over port 3128. You'll need to open this port after you deploy the Console agent.

Enable NTP

If you're planning to use NetApp Data Classification to scan your corporate data sources, you should enable a Network Time Protocol (NTP) service on both the Console agent and the NetApp Data Classification system so that the time is synchronized between the systems. [Learn more about NetApp Data classification](#)

Implement this network access after you create the Console agent.

Step 2: Set up AWS permissions

To prepare for a marketplace deployment, create IAM policies in AWS and attach them to an IAM role. When you create the Console agent from the AWS Marketplace, you are prompted to select that IAM role.

Steps

1. Log in to the AWS console and navigate to the IAM service.
2. Create a policy:
 - a. Select **Policies > Create policy**.
 - b. Select **JSON** and copy and paste the contents of the [IAM policy for the Console agent](#).
 - c. Finish the remaining steps to create the policy.

You may need to create a second policy based on the NetApp data services you plan to use. For standard regions, the permissions are spread across two policies. Two policies are required due to a maximum character size limit for managed policies in AWS. [Learn more about IAM policies for the Console agent](#).

3. Create an IAM role:
 - a. Select **Roles > Create role**.
 - b. Select **AWS service > EC2**.
 - c. Add permissions by attaching the policy that you just created.
 - d. Finish the remaining steps to create the role.

Result

You now have an IAM role that you can associate with the EC2 instance during deployment from the AWS Marketplace.

Step 3: Review instance requirements

When you create the Console agent, you need to choose an EC2 instance type that meets the following requirements.

CPU

8 cores or 8 vCPUs

RAM

32 GB

AWS EC2 instance type

An instance type that meets CPU and RAM requirements. NetApp recommends t3.2xlarge.

Step 4: Create the Console agent

Create the Console agent directly from the AWS Marketplace.

About this task

Creating the Console agent from the AWS Marketplace deploys an EC2 instance in AWS using a default configuration. [Learn about the default configuration for the Console agent.](#)

Before you begin

You should have the following:

- A VPC and subnet that meets networking requirements.
- An IAM role with an attached policy that includes the required permissions for the Console agent.
- Permissions to subscribe and unsubscribe from the AWS Marketplace for your IAM user.
- An understanding of CPU and RAM requirements for the instance.
- A key pair for the EC2 instance.

Steps

1. Go to the [NetApp Console agent listing on the AWS Marketplace](#)
2. On the Marketplace page, select **Continue to Subscribe**.
3. To subscribe to the software, select **Accept Terms**.

The subscription process can take a few minutes.

4. After the subscription process is complete, select **Continue to Configuration**.
5. On the **Configure this software** page, ensure that you've selected the correct region and then select **Continue to Launch**.
6. On the **Launch this software** page, under **Choose Action**, select **Launch through EC2** and then select **Launch**.

Use the EC2 Console to launch the instance and attach an IAM role. This is not possible with the **Launch from Website** action.

7. Follow the prompts to configure and deploy the instance:
 - **Name and tags:** Enter a name and tags for the instance.
 - **Application and OS Images:** Skip this section. The Console agent AMI is already selected.
 - **Instance type:** Depending on region availability, choose an instance type that meets RAM and CPU requirements (t3.2xlarge is preselected and recommended).

- **Key pair (login):** Select the key pair that you want to use to securely connect to the instance.
- **Network settings:** Edit the network settings as needed:
 - Choose the desired VPC and subnet.
 - Specify whether the instance should have a public IP address.
 - Specify security group settings that enable the required connection methods for the Console agent instance: SSH, HTTP, and HTTPS.

[View security group rules for AWS.](#)

- **Configure storage:** Keep the default size and disk type for the root volume.

If you want to enable Amazon EBS encryption on the root volume, select **Advanced**, expand **Volume 1**, select **Encrypted**, and then choose a KMS key.

- **Advanced details:** Under **IAM instance profile**, choose the IAM role that includes the required permissions for the Console agent.
- **Summary:** Review the summary and select **Launch instance**.

AWS launches the Console agent with the specified settings, and the Console agent runs in about ten minutes.



If the installation fails, you can view logs and a report to help you troubleshoot. [Learn how to troubleshoot installation issues.](#)

8. Open a web browser from a host that has a connection to the Console agent virtual machine and URL of the Console agent.
9. After you log in, set up the Console agent:
 - a. Specify the Console organization to associate with the Console agent.
 - b. Enter a name for the system.
 - c. Under **Are you running in a secured environment?** keep restricted mode disabled.

Keep restricted mode disabled to use the Console in standard mode. You should enable restricted mode only if you have a secure environment and want to disconnect this account from the Console backend services. If that's the case, [follow steps to get started with NetApp Console in restricted mode.](#)

- d. Select **Let's start**.

Result

The Console agent is now installed and set up with your Console organization.

Open a web browser and go to the [NetApp Console](#) to start using the Console agent with the Console.

If you have Amazon S3 buckets in the same AWS account where you created the Console agent, you'll see an Amazon S3 working environment appear on the **Systems** page automatically. [Learn how to manage S3 buckets from NetApp Console](#)

Manually install the Console agent in AWS

You can manually install a Console agent on a Linux host running in AWS. To manually install the Console agent on your own Linux host, you need to review host requirements,

set up your networking, prepare AWS permissions, install the Console agent, and then provide the permissions that you prepared.

Before you begin

- You should have an [understanding of Console agents](#).
- You should review [Console agent limitations](#).

Step 1: Review host requirements

Ensure the host running the Console agent software meets operating system, RAM, and port requirements.



The Console agent reserves the UID and GID range of 19000 to 19200. This range is fixed and cannot be modified. If any third-party software on your host is using UIDs or GIDs within this range, the agent installation will fail. NetApp recommends using a host that is free of third-party software to avoid conflicts.

Dedicated host

The Console agent requires a dedicated host. Any architecture is supported if it meets these size requirements:

- CPU: 8 cores or 8 vCPUs
- RAM: 32 GB
- Disk space: 165 GB is recommended for the host, with the following partition requirements:
 - `/opt`: 120 GiB of space must be available

The agent uses `/opt` to install the `/opt/application/netapp` directory and its contents.

- `/var`: 40 GiB of space must be available

The Console agent requires this space in `/var` because Podman or Docker are architected to create the containers within this directory. Specifically, they will create containers in the `/var/lib/containers/storage` directory and `/var/lib/docker` for Docker. External mounts or symlinks do not work for this space.

AWS EC2 instance type

An instance type that meets CPU and RAM requirements. NetApp recommends t3.2xlarge.

Hypervisor

A bare metal or hosted hypervisor that is certified to run a supported operating system is required.

Operating system and container requirements

The Console agent is supported with the following operating systems when using the Console in standard mode or restricted mode. A container orchestration tool is required before you install the agent.

Operating system	Supported OS versions	Supported agent versions	Required container tool	SELinux
Red Hat Enterprise Linux				

Operating system	Supported OS versions	Supported agent versions	Required container tool	SELinux
	<p>9.6</p> <ul style="list-style-type: none"> English language versions only. The host must be registered with Red Hat Subscription Management. If it's not registered, the host can't access repositories to update required 3rd-party software during agent installation. 	4.0.0 or later with the Console in standard mode or restricted mode	<p>Podman version 5.4.0 with podman-compose 1.5.0.</p> <p>View Podman configuration requirements.</p>	Supported in enforcing mode or permissive mode
	<p>9.1 to 9.4</p> <ul style="list-style-type: none"> English language versions only. The host must be registered with Red Hat Subscription Management. If it's not registered, the host can't access repositories to update required 3rd-party software during agent installation. 	3.9.50 or later with the Console in standard mode or restricted mode	<p>Podman version 4.9.4 with podman-compose 1.5.0.</p> <p>View Podman configuration requirements.</p>	Supported in enforcing mode or permissive mode

Operating system	Supported OS versions	Supported agent versions	Required container tool	SELinux
	8.6 to 8.10 <ul style="list-style-type: none"> English language versions only. The host must be registered with Red Hat Subscription Management. If it's not registered, the host can't access repositories to update required 3rd-party software during agent installation. 	3.9.50 or later with the Console in standard mode or restricted mode	Podman version 4.6.1 or 4.9.4 with podman-compose 1.0.6. View Podman configuration requirements.	Supported in enforcing mode or permissive mode
Ubuntu				
	24.04 LTS	3.9.45 or later with the NetApp Console in standard mode or restricted mode	Docker Engine 23.06 to 28.0.0.	Not supported
	22.04 LTS	3.9.50 or later	Docker Engine 23.0.6 to 28.0.0.	Not supported

Key pair

When you create the Console agent, you'll need to select an EC2 key pair to use with the instance.

PUT response hop limit when using IMDSv2

If IMDSv2 is enabled (the default for new EC2 instances), set the PUT response hop limit to 3. If you do not, the system displays a UI initialization error during agent setup.

- [Require the use of IMDSv2 on Amazon EC2 instances](#)
- [AWS documentation: Change the PUT response hop limit](#)

Step 2: Install Podman or Docker Engine

Depending on your operating system, either Podman or Docker Engine is required before installing the agent.

- Podman is required for Red Hat Enterprise Linux 8 and 9.

[View the supported Podman versions.](#)

- Docker Engine is required for Ubuntu.

[View the supported Docker Engine versions.](#)

Example 1. Steps

Podman

Follow these steps to install and configure Podman:

- Enable and start the podman.socket service
- Install python3
- Install the podman-compose package version 1.0.6
- Add podman-compose to the PATH environment variable
- If using Red Hat Enterprise Linux, verify that your Podman version is using Netavark Aardvark DNS instead of CNI



Adjust the aardvark-dns port (default: 53) after installing the agent to avoid DNS port conflicts. Follow the instructions to configure the port.

Steps

1. Remove the podman-docker package if it's installed on the host.

```
dnf remove podman-docker
rm /var/run/docker.sock
```

2. Install Podman.

You can obtain Podman from official Red Hat Enterprise Linux repositories.

- a. For Red Hat Enterprise Linux 9.6:

```
sudo dnf install podman-5:<version>
```

Where <version> is the supported version of Podman that you're installing. [View the supported Podman versions](#).

- b. For Red Hat Enterprise Linux 9.1 to 9.4:

```
sudo dnf install podman-4:<version>
```

Where <version> is the supported version of Podman that you're installing. [View the supported Podman versions](#).

- c. For Red Hat Enterprise Linux 8:

```
sudo dnf install podman-4:<version>
```

Where <version> is the supported version of Podman that you're installing. [View the supported Podman versions](#).

3. Enable and start the podman.socket service.

```
sudo systemctl enable --now podman.socket
```

4. Install python3.

```
sudo dnf install python3
```

5. Install the EPEL repository package if it's not already available on your system.

This step is required because podman-compose is available from the Extra Packages for Enterprise Linux (EPEL) repository.

6. If using Red Hat Enterprise 9:

a. Install the EPEL repository package.

```
sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-9.noarch.rpm
```

a. Install podman-compose package 1.5.0.

```
sudo dnf install podman-compose-1.5.0
```

7. If using Red Hat Enterprise Linux 8:

a. Install the EPEL repository package.

```
sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
```

b. Install podman-compose package 1.0.6.

```
sudo dnf install podman-compose-1.0.6
```



Using the `dnf install` command meets the requirement for adding podman-compose to the PATH environment variable. The installation command adds podman-compose to `/usr/bin`, which is already included in the `secure_path` option on the host.

c. If using Red Hat Enterprise Linux 8, verify that your Podman version is using NetAvark with Aardvark DNS instead of CNI.

i. Check to see if your networkBackend is set to CNI by running the following command:

```
podman info | grep networkBackend
```

- ii. If the `networkBackend` is set to `CNI`, you'll need to change it to `netavark`.
- iii. Install `netavark` and `aardvark-dns` using the following command:

```
dnf install aardvark-dns netavark
```

- iv. Open the `/etc/containers/containers.conf` file and modify the `network_backend` option to use "netavark" instead of "cni".

If `/etc/containers/containers.conf` doesn't exist, make the configuration changes to `/usr/share/containers/containers.conf`.

- v. Restart podman.

```
systemctl restart podman
```

- vi. Confirm `networkBackend` is now changed to "netavark" using the following command:

```
podman info | grep networkBackend
```

Docker Engine

Follow the documentation from Docker to install Docker Engine.

Steps

1. [View installation instructions from Docker](#)

Follow the steps to install a supported Docker Engine version. Do not install the latest version, as it is unsupported by the Console.

2. Verify that Docker is enabled and running.

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

Step 3: Set up networking

Ensure the network location supports the following requirements so the Console agent can manage resources in your hybrid cloud.

Connections to target networks

The Console agent requires a network connection to the location where you're planning to create and manage systems. For example, the network where you plan to create Cloud Volumes ONTAP systems or a

storage system in your on-premises environment.

Outbound internet access

The network location where you deploy the Console agent must have an outbound internet connection to contact specific endpoints.

Endpoints contacted from computers when using the web-based NetApp Console

Computers that access the Console from a web browser must have the ability to contact several endpoints. You'll need to use the Console to set up the Console agent and for day-to-day use of the Console.

[Prepare networking for the NetApp console.](#)

Endpoints contacted from the Console agent

The Console agent requires outbound internet access to contact the following endpoints to manage resources and processes within your public cloud environment for day-to-day operations.

The endpoints listed below are all CNAME entries.

Endpoints	Purpose
AWS services (amazonaws.com): <ul style="list-style-type: none">• CloudFormation• Elastic Compute Cloud (EC2)• Identity and Access Management (IAM)• Key Management Service (KMS)• Security Token Service (STS)• Simple Storage Service (S3)	To manage AWS resources. The endpoint depends on your AWS region. Refer to AWS documentation for details
Amazon FsX for NetApp ONTAP: <ul style="list-style-type: none">• api.workloads.netapp.com	The web-based console contacts this endpoint to interact with the Workload Factory APIs to manage and operate FSx for ONTAP based workloads.
https://mysupport.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support.
https://signin.b2c.netapp.com	To update NetApp Support Site (NSS) credentials or to add new NSS credentials to the NetApp Console.
https://support.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support as well as to receive software updates for Cloud Volumes ONTAP.

Endpoints	Purpose
https://api.blueexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.blueexp.netapp.com https://cdn.auth0.com	To provide features and services within the NetApp Console.
https://blueexpinfraprod.eastus2.data.azurecr.io https://blueexpinfraprod.azurecr.io	<p>To obtain images for Console agent upgrades.</p> <ul style="list-style-type: none"> When you deploy a new agent, the validation check tests connectivity to current endpoints. If you use previous endpoints, the validation check fails. To avoid this failure, skip the validation check. <p>Although the previous endpoints are still supported, NetApp recommends updating your firewall rules to the current endpoints as soon as possible. Learn how to update your endpoint list.</p> <ul style="list-style-type: none"> When you update to the current endpoints in your firewall, your existing agents will continue to work.

Proxy server

NetApp supports both explicit and transparent proxy configurations. If you are using a transparent proxy, you only need to provide the certificate for the proxy server. If you are using an explicit proxy, you'll also need the IP address and credentials.

- IP address
- Credentials
- HTTPS certificate

Ports

There's no incoming traffic to the Console agent, unless you initiate it or if it is used as a proxy to send AutoSupport messages from Cloud Volumes ONTAP to NetApp Support.

- HTTP (80) and HTTPS (443) provide access to the local UI, which you'll use in rare circumstances.
- SSH (22) is only needed if you need to connect to the host for troubleshooting.
- Inbound connections over port 3128 are required if you deploy Cloud Volumes ONTAP systems in a subnet where an outbound internet connection isn't available.

If Cloud Volumes ONTAP systems don't have an outbound internet connection to send AutoSupport messages, the Console automatically configures those systems to use a proxy server that's included with the Console agent. The only requirement is to ensure that the Console agent's security group allows inbound connections over port 3128. You'll need to open this port after you deploy the Console agent.

Enable NTP

If you're planning to use NetApp Data Classification to scan your corporate data sources, you should enable a Network Time Protocol (NTP) service on both the Console agent and the NetApp Data Classification system so that the time is synchronized between the systems. [Learn more about NetApp Data classification](#)

Step 4: Set up AWS permissions for the Console

Provide AWS permissions to the NetApp Console using one of these options:

- Option 1: Create IAM policies and attach the policies to an IAM role that you can associate with the EC2 instance.
- Option 2: Provide the Console with the AWS access key for an IAM user who has the required permissions.

Follow the steps to prepare permissions for the Console.

IAM role

Steps

1. Log in to the AWS console and navigate to the IAM service.
2. Create a policy:
 - a. Select **Policies > Create policy**.
 - b. Select **JSON** and copy and paste the contents of the [IAM policy for the Console agent](#).
 - c. Finish the remaining steps to create the policy.

Depending on the NetApp data services you plan to use, you might need to create a second policy. For standard regions, the permissions are spread across two policies. Two policies are required due to a maximum character size limit for managed policies in AWS. [Learn more about IAM policies for the Console agent](#).

3. Create an IAM role:
 - a. Select **Roles > Create role**.
 - b. Select **AWS service > EC2**.
 - c. Add permissions by attaching the policy that you just created.
 - d. Finish the remaining steps to create the role.

Result

You now have an IAM role that you can associate with the EC2 instance after you install the Console agent.

AWS access key

Steps

1. Log in to the AWS console and navigate to the IAM service.
2. Create a policy:
 - a. Select **Policies > Create policy**.
 - b. Select **JSON** and copy and paste the contents of the [IAM policy for the Console agent](#).
 - c. Finish the remaining steps to create the policy.

Depending on the NetApp data services that you plan to use, you might need to create a second policy.

For standard regions, the permissions are spread across two policies. Two policies are required due to a maximum character size limit for managed policies in AWS. [Learn more about IAM policies for the Console agent](#).

3. Attach the policies to an IAM user.
 - [AWS Documentation: Creating IAM Roles](#)
 - [AWS Documentation: Adding and Removing IAM Policies](#)
4. Ensure that the user has an access key that you can add to the NetApp Console after you install the Console agent.

Result

You now have an IAM user that has the required permissions and an access key that you can provide to the Console.

Step 5: Install the Console agent

After you complete the prerequisites, manually install the software on your Linux host.

Before you begin

You should have the following:

- Root privileges to install the Console agent.
- Details about a proxy server, if a proxy is required for internet access from the Console agent.

You have the option to configure a proxy server after installation but doing so requires restarting the Console agent.

- A CA-signed certificate, if the proxy server uses HTTPS or if the proxy is an intercepting proxy.



You cannot set a certificate for a transparent proxy server when manually installing the Console agent. If you need to set a certificate for a transparent proxy server, you must use the Maintenance Console after installation. Learn more about the [Agent Maintenance Console](#).

About this task

After installation, the Console agent automatically updates itself if a new version is available.

Steps

1. If the `http_proxy` or `https_proxy` system variables are set on the host, remove them:

```
unset http_proxy
unset https_proxy
```

If you don't remove these system variables, the installation fails.

2. Download the Console agent software and then copy it to the Linux host. You can download it either from the NetApp Console or the NetApp Support site.

- NetApp Console: Go to **Agents > Management > Deploy agent > On-prem > Manual install**.

Choose download the agent installer files or a URL to the files.

- NetApp Support Site (needed if you don't already have access to the Console) [NetApp Support Site](#),

3. Assign permissions to run the script.

```
chmod +x NetApp_Console_Agent_Cloud_<version>
```

Where `<version>` is the version of the Console agent that you downloaded.

4. If installing in a Government Cloud environment, disable the configuration checks. [Learn how to disable configuration checks for manual installations](#).

5. Run the installation script.

```
./NetApp_Console_Agent_Cloud_<version> --proxy <HTTP or HTTPS proxy server> --cacert <path and file name of a CA-signed certificate>
```

You'll need to add proxy information if your network requires a proxy for internet access. You can add an explicit proxy during installation. The `--proxy` and `--cacert` parameters are optional and you won't be prompted to add them. If you have an explicit proxy server, you will need to enter the parameters as shown.



If you want to configure a transparent proxy, you can do so after you've installed. [Learn about the agent maintenance console](#)

+

Here is an example configuring an explicit proxy server with a CA-signed certificate:

+

```
./NetApp_Console_Agent_Cloud_v4.0.0--proxy  
https://user:password@10.0.0.30:8080/ --cacert /tmp/cacert/certificate.cer
```

+

`--proxy` configures the Console agent to use an HTTP or HTTPS proxy server using one of the following formats:

+

- * http://address:port
- * http://user-name:password@address:port
- * http://domain-name%92user-name:password@address:port
- * https://address:port
- * https://user-name:password@address:port
- * https://domain-name%92user-name:password@address:port

+

Note the following:

+

The user can be a local user or domain user.

For a domain user, you must use the ASCII code for a \ as shown above.

The Console agent doesn't support user names or passwords that include the @ character.

If the password includes any of the following special characters, you must escape that special character by prepending it with a backslash: & or !

+

For example:

+

http://bxpproxyuser:netapp1\!@address:3128

1. If you used Podman, you'll need to adjust the aardvark-dns port.
 - a. SSH to the Console agent virtual machine.
 - b. Open podman `/usr/share/containers/containers.conf` file and modify the chosen port for Aardvark DNS service. For example, change it to 54.

```
vi /usr/share/containers/containers.conf
```

For example:

```
# Port to use for dns forwarding daemon with netavark in rootful
bridge
# mode and dns enabled.
# Using an alternate port might be useful if other DNS services
should
# run on the machine.
#
dns_bind_port = 54
```

- c. Reboot the Console agent virtual machine.
2. Wait for the installation to complete.

At the end of the installation, the Console agent service (occm) restarts twice if you specified a proxy server.



If the installation fails, you can view the installation report and logs to help you fix the issues.
[Learn how to troubleshoot installation issues.](#)

1. Open a web browser from a host that has a connection to the Console agent virtual machine and enter the following URL:

`https://ipaddress`

2. After you log in, set up the Console agent:
 - a. Specify the organization to associate with the Console agent.
 - b. Enter a name for the system.
 - c. Under **Are you running in a secured environment?** keep restricted mode disabled.

You should keep restricted mode disabled because these steps describe how to use the Console in standard mode. You should enable restricted mode only if you have a secure environment and want to disconnect this account from backend services. If that's the case, [follow steps to get started with the NetApp Console in restricted mode.](#)

- d. Select **Let's start**.

If you have Amazon S3 buckets in the same AWS account where you created the Console agent, you'll see an Amazon S3 storage system appear on the **Systems** page automatically. [Learn how to manage S3 buckets](#)

Step 6: Provide permissions to NetApp Console

After you install the Console agent, provide the AWS permissions you set up so the Console agent can manage your data and storage infrastructure in AWS.

IAM role

Attach the IAM role you create to the Console agent EC2 instance.

Steps

1. Go to the Amazon EC2 console.
2. Select **Instances**.
3. Select the Console agent instance.
4. Select **Actions > Security > Modify IAM role**.
5. Select the IAM role and select **Update IAM role**.

Go to the [NetApp Console](#) to start using the Console agent.

AWS access key

Provide the Console with the AWS access key for an IAM user that has the required permissions.

Steps

1. Ensure that the correct Console agent is currently selected in the Console.
2. Select **Administration > Credentials**.
3. Select **Organization credentials**.
4. Select **Add Credentials** and follow the steps in the wizard.
 - a. **Credentials Location**: Select *Amazon Web Services > Agent.
 - b. **Define Credentials**: Enter an AWS access key and secret key.
 - c. **Marketplace Subscription**: Associate a Marketplace subscription with these credentials by subscribing now or by selecting an existing subscription.
 - d. **Review**: Confirm the details about the new credentials and select **Add**.

Go to the [NetApp Console](#) to start using the Console agent.

Azure

Console agent installation options in Azure

There are a few different ways to create a Console agent in Azure. Directly from the NetApp Console is the most common way.

The following installation options are available:

- [Create a Console agent directly from the NetApp Console](#) (this is the standard option)

This action launches a VM running Linux and the Console agent software in a VNet of your choice.

- [Create a Console agent from the Azure Marketplace](#)

This action also launches a VM running Linux and the Console agent software, but the deployment is initiated directly from the Azure Marketplace, rather than from the Console.

- [Download and manually install the software on your own Linux host](#)

The installation option that you choose impacts how you prepare for installation. This includes how you provide the Console agent with the required permissions that it needs to authenticate and manage resources in Azure.

Create a Console agent in Azure from NetApp Console

To create a Console agent in Azure from the NetApp Console, you need to set up your networking, prepare Azure permissions, and then create the Console agent.

Before you begin

- You should have an [understanding of Console agents](#).
- You should review [Console agent limitations](#).

Step 1: Set up networking

Ensure that the network location where you plan to install the Console agent supports the following requirements. These requirements allow the Console agent to manage hybrid cloud resources.

Azure region

If you use Cloud Volumes ONTAP, the Console agent 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](#)

VNet and subnet

When you create the Console agent, you need to specify the VNet and subnet where it should reside.

Connections to target networks

The Console agent requires a network connection to the location where you're planning to create and manage systems. For example, the network where you plan to create Cloud Volumes ONTAP systems or a storage system in your on-premises environment.

Outbound internet access

The network location where you deploy the Console agent must have an outbound internet connection to contact specific endpoints.

Endpoints contacted from the Console agent

The Console agent requires outbound internet access to contact the following endpoints to manage resources and processes within your public cloud environment for day-to-day operations.

The endpoints listed below are all CNAME entries.

Endpoints	Purpose
https://management.azure.com https://login.microsoftonline.com https://blob.core.windows.net https://core.windows.net	To manage resources in Azure public regions.
https://management.chinacloudapi.cn https://login.chinacloudapi.cn https://blob.core.chinacloudapi.cn https://core.chinacloudapi.cn	To manage resources in Azure China regions.
https://mysupport.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support.
https://signin.b2c.netapp.com	To update NetApp Support Site (NSS) credentials or to add new NSS credentials to the NetApp Console.
https://support.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support as well as to receive software updates for Cloud Volumes ONTAP.
https://api.blueexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.blueexp.netapp.com https://cdn.auth0.com	To provide features and services within the NetApp Console.
https://blueexpinfraprod.eastus2.data.azurecr.io https://blueexpinfraprod.azurecr.io	<p>To obtain images for Console agent upgrades.</p> <ul style="list-style-type: none"> When you deploy a new agent, the validation check tests connectivity to current endpoints. If you use previous endpoints, the validation check fails. To avoid this failure, skip the validation check. <p>Although the previous endpoints are still supported, NetApp recommends updating your firewall rules to the current endpoints as soon as possible. Learn how to update your endpoint list.</p> <ul style="list-style-type: none"> When you update to the current endpoints in your firewall, your existing agents will continue to work.

Endpoints contacted from the NetApp console

As you use the web-based NetApp Console that's provided through the SaaS layer, it contacts several endpoints to complete data management tasks. This includes endpoints that are contacted to deploy the Console agent from the the Console.

[View the list of endpoints contacted from the NetApp console.](#)

Proxy server

NetApp supports both explicit and transparent proxy configurations. If you are using a transparent proxy, you only need to provide the certificate for the proxy server. If you are using an explicit proxy, you'll also need the IP address and credentials.

- IP address
- Credentials
- HTTPS certificate

Ports

There's no incoming traffic to the Console agent, unless you initiate it or if it is used as a proxy to send AutoSupport messages from Cloud Volumes ONTAP to NetApp Support.

- HTTP (80) and HTTPS (443) provide access to the local UI, which you'll use in rare circumstances.
- SSH (22) is only needed if you need to connect to the host for troubleshooting.
- Inbound connections over port 3128 are required if you deploy Cloud Volumes ONTAP systems in a subnet where an outbound internet connection isn't available.

If Cloud Volumes ONTAP systems don't have an outbound internet connection to send AutoSupport messages, the Console automatically configures those systems to use a proxy server that's included with the Console agent. The only requirement is to ensure that the Console agent's security group allows inbound connections over port 3128. You'll need to open this port after you deploy the Console agent.

Enable NTP

If you're planning to use NetApp Data Classification to scan your corporate data sources, you should enable a Network Time Protocol (NTP) service on both the Console agent and the NetApp Data Classification system so that the time is synchronized between the systems. [Learn more about NetApp Data classification](#)

You need to implement this networking requirement after you create the Console agent.

Step 2: Create a Console agent deployment policy (custom role)

You need to create a custom role that has permissions to deploy the Console agent in Azure.

Create an Azure custom role that you can assign to your Azure account or to a Microsoft Entra service principal. The Console authenticates with Azure and uses these permissions to create the Console agent on your behalf.

The Console deploys the Console agent VM in Azure, enables a [system-assigned managed identity](#), creates the required role, and assigns it to the VM. [Review how the Console uses the permissions.](#)

Note that you can create an Azure custom role using the Azure portal, Azure PowerShell, Azure CLI, or REST API. The following steps show how to create the role using the Azure CLI. If you would prefer to use a different method, refer to [Azure documentation](#)

Steps

1. Copy the required permissions for a new custom role in Azure and save them in a JSON file.



This custom role contains only the permissions needed to launch the Console agent VM in Azure from the Console. Don't use this policy for other situations. When the Console creates the Console agent, it applies a new set of permissions to the Console agent VM that enables the Console agent to manage Azure resources.

```
{
  "Name": "Azure SetupAsService",
  "Actions": [
    "Microsoft.Compute/disks/delete",
    "Microsoft.Compute/disks/read",
    "Microsoft.Compute/disks/write",
    "Microsoft.Compute/locations/operations/read",
    "Microsoft.Compute/operations/read",
    "Microsoft.Compute/virtualMachines/instanceView/read",
    "Microsoft.Compute/virtualMachines/read",
    "Microsoft.Compute/virtualMachines/write",
    "Microsoft.Compute/virtualMachines/delete",
    "Microsoft.Compute/virtualMachines/extensions/write",
    "Microsoft.Compute/virtualMachines/extensions/read",
    "Microsoft.Compute/availabilitySets/read",
    "Microsoft.Network/locations/operationResults/read",
    "Microsoft.Network/locations/operations/read",
    "Microsoft.Network/networkInterfaces/join/action",
    "Microsoft.Network/networkInterfaces/read",
    "Microsoft.Network/networkInterfaces/write",
    "Microsoft.Network/networkInterfaces/delete",
    "Microsoft.Network/networkSecurityGroups/join/action",
    "Microsoft.Network/networkSecurityGroups/read",
    "Microsoft.Network/networkSecurityGroups/write",
    "Microsoft.Network/virtualNetworks/checkIpAddressAvailability/read",
    "Microsoft.Network/virtualNetworks/read",
    "Microsoft.Network/virtualNetworks/subnets/join/action",
    "Microsoft.Network/virtualNetworks/subnets/read",
    "Microsoft.Network/virtualNetworks/subnets/virtualMachines/read",
    "Microsoft.Network/virtualNetworks/virtualMachines/read",
    "Microsoft.Network/publicIPAddresses/write",
    "Microsoft.Network/publicIPAddresses/read",
    "Microsoft.Network/publicIPAddresses/delete",
    "Microsoft.Network/networkSecurityGroups/securityRules/read",
    "Microsoft.Network/networkSecurityGroups/securityRules/write",
    "Microsoft.Network/networkSecurityGroups/securityRules/delete",
    "Microsoft.Network/publicIPAddresses/join/action",

    "Microsoft.Network/locations/virtualNetworkAvailableEndpointServices/read",
    "Microsoft.Network/networkInterfaces/ipConfigurations/read",
```

```

    "Microsoft.Resources/deployments/operations/read",
    "Microsoft.Resources/deployments/read",
    "Microsoft.Resources/deployments/delete",
    "Microsoft.Resources/deployments/cancel/action",
    "Microsoft.Resources/deployments/validate/action",
    "Microsoft.Resources/resources/read",
    "Microsoft.Resources/subscriptions/operationresults/read",
    "Microsoft.Resources/subscriptions/resourceGroups/delete",
    "Microsoft.Resources/subscriptions/resourceGroups/read",
    "Microsoft.Resources/subscriptions/resourcegroups/resources/read",
    "Microsoft.Resources/subscriptions/resourceGroups/write",
    "Microsoft.Authorization/roleDefinitions/write",
    "Microsoft.Authorization/roleAssignments/write",

    "Microsoft.MarketplaceOrdering/offertypes/publishers/offers/plans/agreements/read",

    "Microsoft.MarketplaceOrdering/offertypes/publishers/offers/plans/agreements/write",
    "Microsoft.Network/networkSecurityGroups/delete",
    "Microsoft.Storage/storageAccounts/delete",
    "Microsoft.Storage/storageAccounts/write",
    "Microsoft.Resources/deployments/write",
    "Microsoft.Resources/deployments/operationStatuses/read",
    "Microsoft.Authorization/roleAssignments/read"
  ],
  "NotActions": [],
  "AssignableScopes": [],
  "Description": "Azure SetupAsService",
  "IsCustom": "true"
}

```

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

Example

```

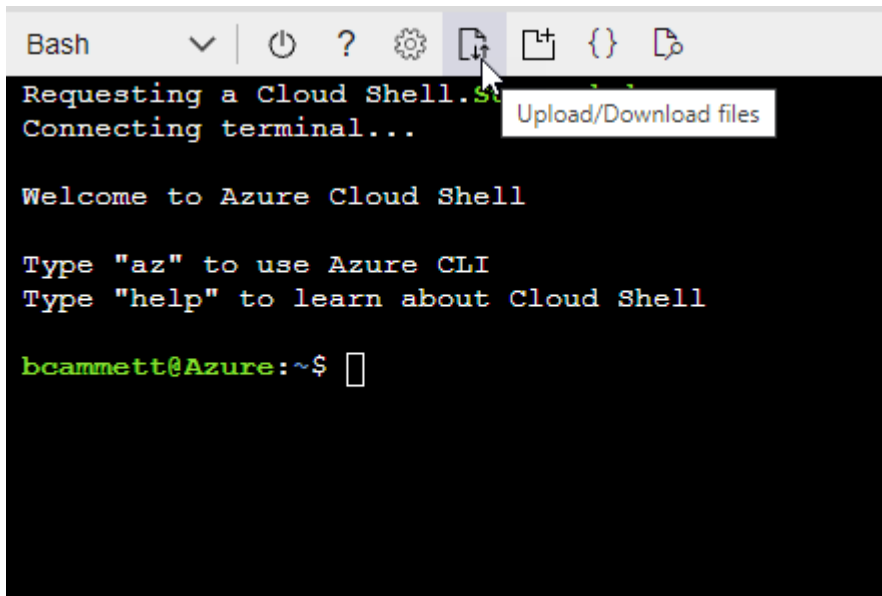
"AssignableScopes": [
  "/subscriptions/d333af45-0d07-4154-943d-c25fbzzzzzzz"
]

```

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 now have a custom role called *Azure SetupAsService*. You can apply this custom role to your user account or to a service principal.

Step 3: Set up authentication

When creating the Console agent from the Console, you need to provide a login that enables the Console to authenticate with Azure and deploy the VM. You have two options:

1. Sign in with your Azure account when prompted. This account must have specific Azure permissions. This is the default option.
2. Provide details about a Microsoft Entra service principal. This service principal also requires specific permissions.

Follow the steps to prepare one of these authentication methods for use with the Console.

Azure account

Assign the custom role to the user who will deploy the Console agent from the Console.

Steps

1. In the Azure portal, open the **Subscriptions** service and select the user's subscription.
2. Click **Access control (IAM)**.
3. Click **Add > Add role assignment** and then add the permissions:
 - a. Select the **Azure SetupAsService** role and click **Next**.



Azure SetupAsService is the default name provided in the Console agent deployment policy for Azure. If you chose a different name for the role, then select that name instead.

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

Service principal

Rather than logging in with your Azure account, you can provide the Console with the credentials for an Azure service principal that has the required permissions.

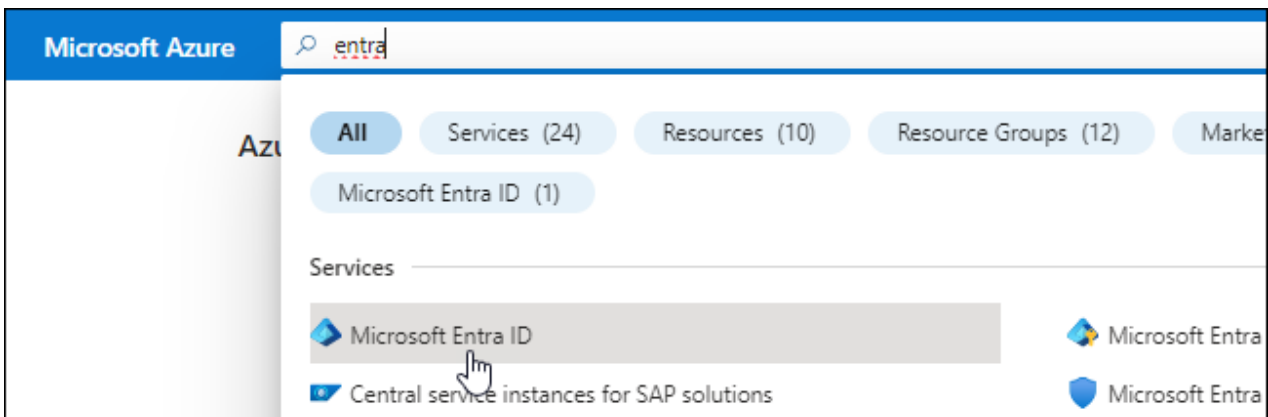
Create and set up a service principal in Microsoft Entra ID and obtain the Azure credentials that the Console needs.

Create a Microsoft Entra application for role-based access control

1. Ensure that you have 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](#)

2. From the Azure portal, open the **Microsoft Entra ID** service.



3. In the menu, select **App registrations**.
4. Select **New registration**.

5. Specify details about the application:

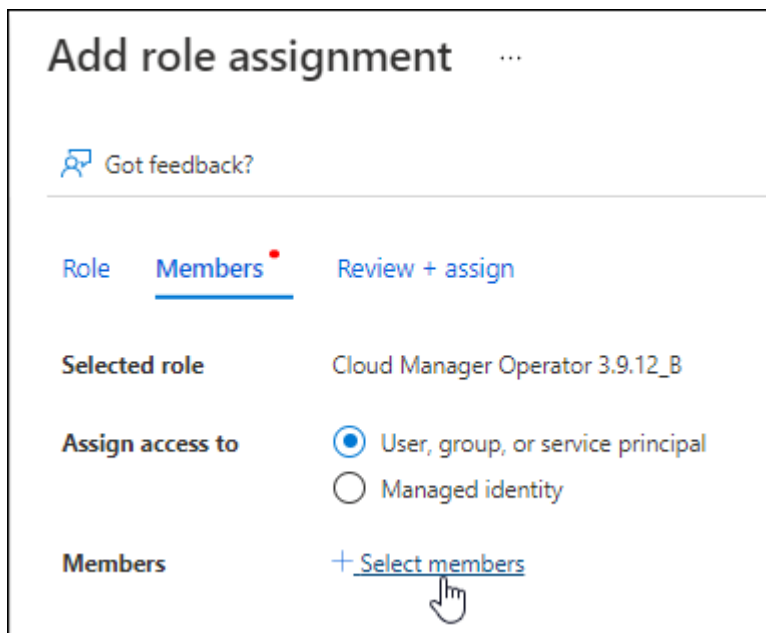
- **Name:** Enter a name for the application.
- **Account type:** Select an account type (any will work with the NetApp Console).
- **Redirect URI:** You can leave this field blank.

6. Select **Register**.


You've created the AD application and service principal.

Assign the custom role to the application

1. From the Azure portal, open the **Subscriptions** service.
2. Select the subscription.
3. Click **Access control (IAM) > Add > Add role assignment**.
4. In the **Role** tab, select the **Console Operator** role and click **Next**.
5. In the **Members** tab, complete the following steps:
 - a. Keep **User, group, or service principal** selected.
 - b. Click **Select members**.



Add role assignment ...

 Got feedback?

Role **Members** [•] [Review + assign](#)

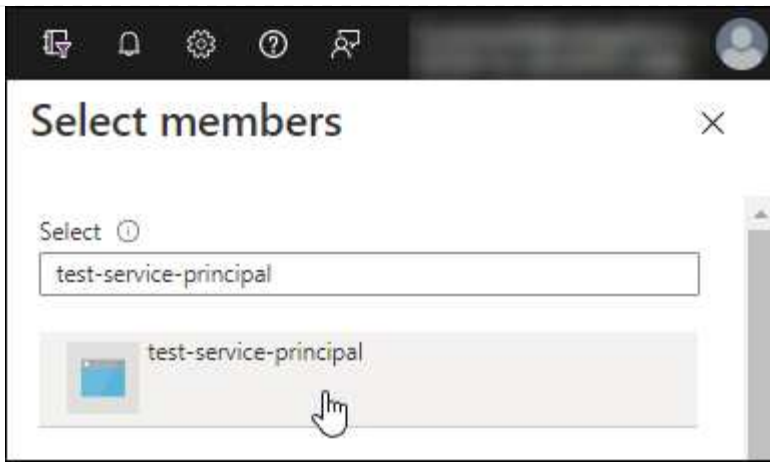
Selected role Cloud Manager Operator 3.9.12_B

Assign access to ☒ User, group, or service principal
 ☐ Managed identity

Members [+ Select members](#)

- c. Search for the name of the application.

Here's an example:



- d. Select the application and click **Select**.
- e. Click **Next**.
6. Click **Review + assign**.

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

If you want to manage resources in multiple Azure subscriptions, then you must bind the service principal to each of those subscriptions. For example, the Console enables you to select the subscription that you want to use when deploying Cloud Volumes ONTAP.

Add Windows Azure Service Management API permissions

1. In the **Microsoft Entra ID** service, select **App registrations** and select the application.
2. Select **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. Select **Access Azure Service Management as organization users** and then select **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

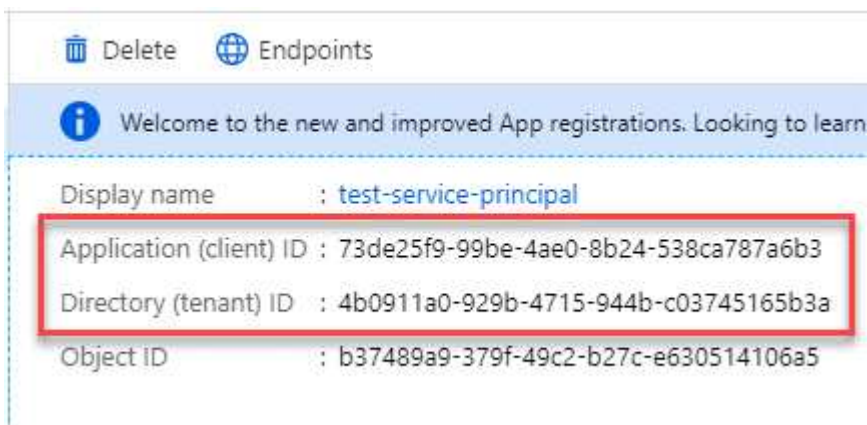


user_impersonation

Access Azure Service Management as organization users (preview) ⓘ

Get the application ID and directory ID for the application

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



When you add the Azure account to the Console, you need to provide the application (client) ID and the directory (tenant) ID for the application. The Console uses the IDs to programmatically sign in.

Create a client secret

1. Open the **Microsoft Entra ID** service.
2. Select **App registrations** and select your application.
3. Select **Certificates & secrets > New client secret**.
4. Provide a description of the secret and a duration.
5. Select **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	Copy to clipboard
test secret	8/16/2020	*sZ1jSe2By:D*-ZRoV4NLfdAcY7:+0vA	

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 the Console when you create the Console agent.

Step 4: Create the Console agent

Create the Console agent directly from the NetApp Console.

About this task

- Creating the Console agent from the Console deploys a virtual machine in Azure using a default configuration. Do not switch to a smaller VM instance with fewer CPUs or less RAM after creating the Console agent. [Learn about the default configuration for the Console agent.](#)
- When the Console deploys the Console agent, it creates a custom role and assigns it to the Console agent VM. This role includes permissions that enables the Console agent to manage Azure resources. You need to ensure that the role is kept up to date as new permissions are added in subsequent releases. [Learn more about the custom role for the Console agent.](#)

Before you begin

You should have the following:

- An Azure subscription.
- A VNet and subnet in your Azure region of choice.
- Details about a proxy server, if your organization requires a proxy for all outgoing internet traffic:
 - IP address
 - Credentials
 - HTTPS certificate
- An SSH public key, if you want to use that authentication method for the Console agent virtual machine. The other option for the authentication method is to use a password.

[Learn about connecting to a Linux VM in Azure](#)

- If you don't want the Console to automatically create an Azure role for the Console agent, then you'll need to create your own [using the policy on this page.](#)

These permissions are for the Console agent itself. It's a different set of permissions than what you previously set up to deploy the Console agent VM.

Steps

1. Select **Administration > Agents**.
2. On the **Overview** page, select **Deploy agent > Azure**
3. On the **Review** page, review the requirements for deploying an agent. Those requirements are also detailed above on this page.
4. On the **Virtual Machine Authentication** page, select the authentication option that matches how you set up Azure permissions:

- Select **Log in** to log in to your Microsoft account, which should have the required permissions.

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 the Console automatically uses that account. If you have multiple accounts, then you might need to log out first to ensure that you're using the right account.

- Select **Active Directory service principal** to enter information about the Microsoft Entra service principal that grants the required permissions:
 - Application (client) ID
 - Directory (tenant) ID
 - Client Secret

[Learn how to obtain these values for a service principal.](#)

5. On the **Virtual Machine Authentication** page, choose an Azure subscription, a location, a new resource group or an existing resource group, and then choose an authentication method for the Console agent virtual machine that you're creating.

The authentication method for the virtual machine can be a password or an SSH public key.

[Learn about connecting to a Linux VM in Azure](#)

6. On the **Details** page, enter a name for the agent, specify tags, and choose whether you want the Console 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 Azure subscriptions associated with this role. Each subscription that you choose provides the Console agent permissions to manage resources in that subscription (for example, Cloud Volumes ONTAP).

7. On the **Network** page, choose a VNet and subnet, whether to enable a public IP address, and optionally specify a proxy configuration.
 - On the **Security Group** page, choose whether to create a new security group or whether to select an existing security group that allows the required inbound and outbound rules.

[View security group rules for Azure.](#)

8. Review your selections to verify that your set up is correct.
 - a. The **Validate agent configuration** check box is marked by default to have the Console validate the network connectivity requirements when you deploy. If the Console fails to deploy the agent, it provides a report to help you troubleshoot. If the deployment succeeds, no report is provided.

If you are still using the [previous endpoints](#) used for agent upgrades, the validation fails with an error. To avoid this, unmark the check box to skip the validation check.

9. Select **Add**.

The Console prepares the agent in about 10 minutes. Stay on the page until the process completes.

Result

After the process is complete, the Console agent is available for use from the Console.



If the deployment fails, you can download a report and logs from the Console to help you fix the issues. [Learn how to troubleshoot installation issues.](#)

If you have Azure Blob storage in the same Azure account where you created the Console agent, you'll see Azure Blob storage appear on the **Systems** page automatically. [Learn how to manage Azure Blob storage from NetApp Console](#)

Create a Console agent from the Azure Marketplace

You can create a Console agent in Azure directly from the Azure Marketplace. To create a Console agent from the Azure Marketplace, you need to set up your networking, prepare Azure permissions, review instance requirements, and then create the Console agent.

Before you begin

- You should have an [understanding of Console agents](#).
- Review [Console agent limitations](#).

Step 1: Set up networking

Ensure that the network location where you plan to install the Console agent supports the following requirements. These requirements enable the Console agent to manage resources in your hybrid cloud.

Azure region

If you use Cloud Volumes ONTAP, the Console agent 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](#)

VNet and subnet

When you create the Console agent, you need to specify the VNet and subnet where it should reside.

Connections to target networks

The Console agent requires a network connection to the location where you're planning to create and manage systems. For example, the network where you plan to create Cloud Volumes ONTAP systems or a storage system in your on-premises environment.

Outbound internet access

The network location where you deploy the Console agent must have an outbound internet connection to contact specific endpoints.

Endpoints contacted from the Console agent

The Console agent requires outbound internet access to contact the following endpoints to manage resources and processes within your public cloud environment for day-to-day operations.

The endpoints listed below are all CNAME entries.

Endpoints	Purpose
https://management.azure.com https://login.microsoftonline.com https://blob.core.windows.net https://core.windows.net	To manage resources in Azure public regions.
https://management.chinacloudapi.cn https://login.chinacloudapi.cn https://blob.core.chinacloudapi.cn https://core.chinacloudapi.cn	To manage resources in Azure China regions.
https://mysupport.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support.
https://signin.b2c.netapp.com	To update NetApp Support Site (NSS) credentials or to add new NSS credentials to the NetApp Console.
https://support.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support as well as to receive software updates for Cloud Volumes ONTAP.
https://api.bluexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.bluexp.netapp.com https://cdn.auth0.com	To provide features and services within the NetApp Console.

Endpoints	Purpose
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	<p>To obtain images for Console agent upgrades.</p> <ul style="list-style-type: none"> When you deploy a new agent, the validation check tests connectivity to current endpoints. If you use previous endpoints, the validation check fails. To avoid this failure, skip the validation check. <p>Although the previous endpoints are still supported, NetApp recommends updating your firewall rules to the current endpoints as soon as possible. Learn how to update your endpoint list.</p> <ul style="list-style-type: none"> When you update to the current endpoints in your firewall, your existing agents will continue to work.

Proxy server

NetApp supports both explicit and transparent proxy configurations. If you are using a transparent proxy, you only need to provide the certificate for the proxy server. If you are using an explicit proxy, you'll also need the IP address and credentials.

- IP address
- Credentials
- HTTPS certificate

Ports

There's no incoming traffic to the Console agent, unless you initiate it or if it is used as a proxy to send AutoSupport messages from Cloud Volumes ONTAP to NetApp Support.

- HTTP (80) and HTTPS (443) provide access to the local UI, which you'll use in rare circumstances.
- SSH (22) is only needed if you need to connect to the host for troubleshooting.
- Inbound connections over port 3128 are required if you deploy Cloud Volumes ONTAP systems in a subnet where an outbound internet connection isn't available.

If Cloud Volumes ONTAP systems don't have an outbound internet connection to send AutoSupport messages, the Console automatically configures those systems to use a proxy server that's included with the Console agent. The only requirement is to ensure that the Console agent's security group allows inbound connections over port 3128. You'll need to open this port after you deploy the Console agent.

Enable NTP

If you're planning to use NetApp Data Classification to scan your corporate data sources, you should enable a Network Time Protocol (NTP) service on both the Console agent and the NetApp Data Classification system so that the time is synchronized between the systems. [Learn more about NetApp Data classification](#)

Implement the networking requirements after creating the Console agent.

Step 2: Review VM requirements

When you create the Console agent, choose a virtual machine type that meets the following requirements.

CPU

8 cores or 8 vCPUs

RAM

32 GB

Azure VM size

An instance type that meets CPU and RAM requirements. NetApp recommends Standard_D8s_v3.

Step 3: Set up permissions

You can provide permissions in the following ways:

- Option 1: Assign a custom role to the Azure VM using a system-assigned managed identity.
- Option 2: Provide the Console with the credentials for an Azure service principal that has the required permissions.

Follow these steps to set up permissions for the Console.

Custom role

Note that you can create an Azure custom role using the Azure portal, Azure PowerShell, Azure CLI, or REST API. The following steps show how to create the role using the Azure CLI. If you would prefer to use a different method, refer to [Azure documentation](#)

Steps

1. If you're planning to manually install the software on your own host, enable a system-assigned managed identity on the VM so that you can provide the required Azure permissions through a custom role.

[Microsoft Azure documentation: Configure managed identities for Azure resources on a VM using the Azure portal](#)

2. Copy the contents of the [custom role permissions for the Connector](#) and save them in a JSON file.
3. Modify the JSON file by adding Azure subscription IDs to the assignable scope.

You should add the ID for each Azure subscription that you want to use with the NetApp Console.

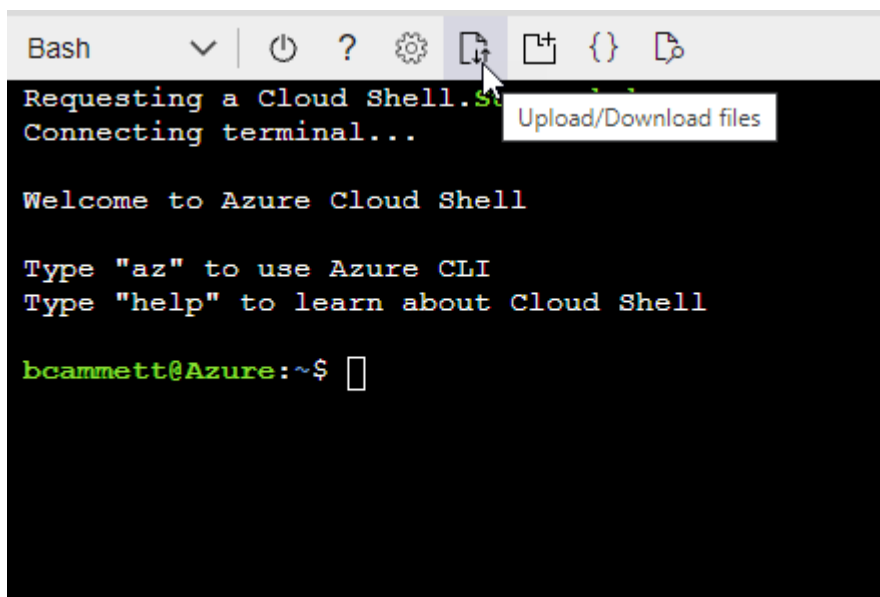
Example

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

4. 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. Use the Azure CLI to create the custom role:

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

Service principal

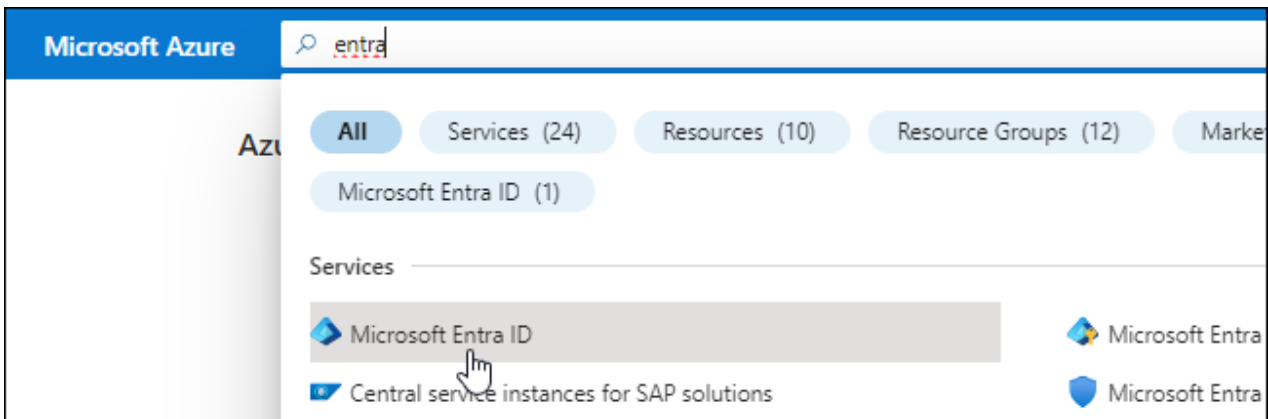
Create and set up a service principal in Microsoft Entra ID and obtain the Azure credentials that the Console needs.

Create a Microsoft Entra application for role-based access control

1. Ensure that you have 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](#)

2. From the Azure portal, open the **Microsoft Entra ID** service.



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

You've created the AD application and service principal.

Assign the application to a role

1. Create a custom role:

Note that you can create an Azure custom role using the Azure portal, Azure PowerShell, Azure CLI, or REST API. The following steps show how to create the role using the Azure CLI. If you would prefer to use a different method, refer to [Azure documentation](#)

- a. Copy the contents of the [custom role permissions for the Console agent](#) and save them in a JSON file.

- 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.

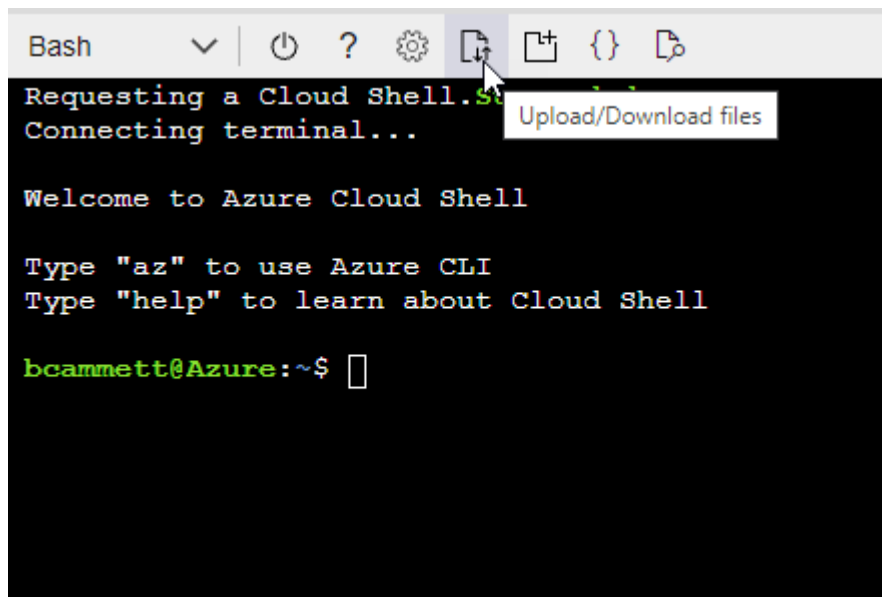
Example

```
"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 steps describe how to create the role by using Bash in Azure Cloud Shell.

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



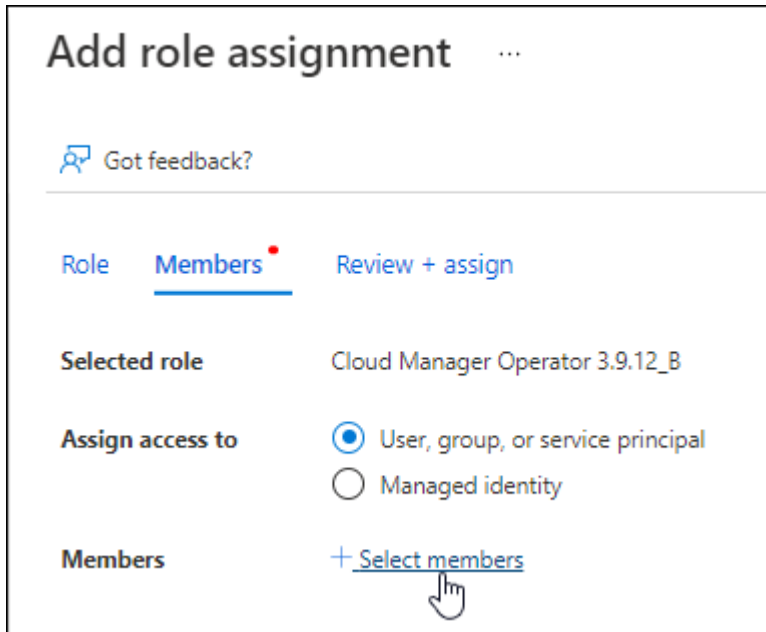
- Use the Azure CLI to create the custom role:

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

You should now have a custom role called Console Operator that you can assign to the Console agent virtual machine.

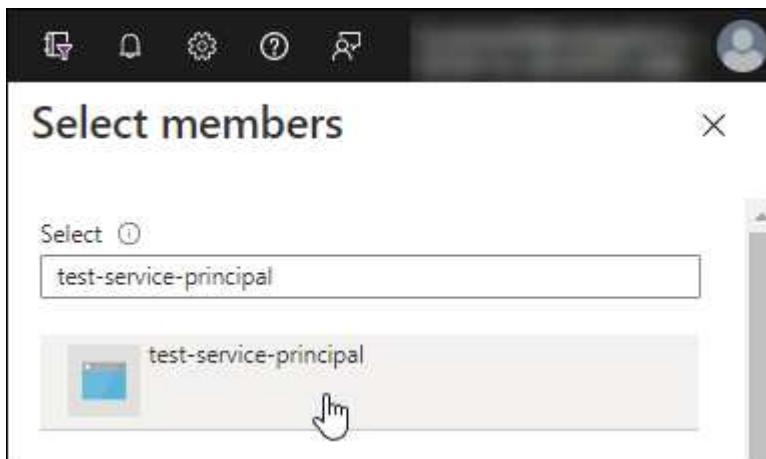
2. Assign the application to the role:
 - a. From the Azure portal, open the **Subscriptions** service.
 - b. Select the subscription.

- c. Select **Access control (IAM) > Add > Add role assignment**.
- d. In the **Role** tab, select the **Console Operator** role and select **Next**.
- e. In the **Members** tab, complete the following steps:
 - Keep **User, group, or service principal** selected.
 - Select **Select members**.



- Search for the name of the application.

Here's an example:



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

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

If you want to deploy Cloud Volumes ONTAP from multiple Azure subscriptions, then you must bind the service principal to each of those subscriptions. In the NetApp Console, you can select

the subscription that you want to use when deploying Cloud Volumes ONTAP.

Add Windows Azure Service Management API permissions

1. In the **Microsoft Entra ID** service, select **App registrations** and select the application.
2. Select **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. Select **Access Azure Service Management as organization users** and then select **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

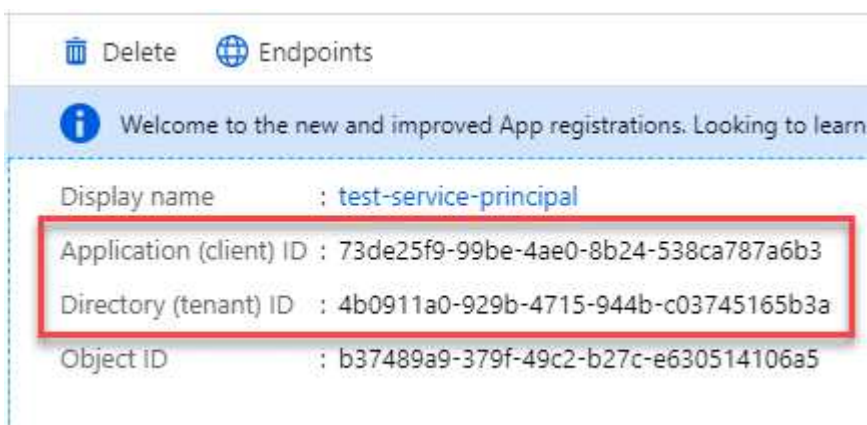


user_impersonation

Access Azure Service Management as organization users (preview) ⓘ

Get the application ID and directory ID for the application

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



When you add the Azure account to the Console, you need to provide the application (client) ID and the directory (tenant) ID for the application. The Console uses the IDs to programmatically sign in.

Create a client secret

1. Open the **Microsoft Entra ID** service.
2. Select **App registrations** and select your application.
3. Select **Certificates & secrets > New client secret**.
4. Provide a description of the secret and a duration.
5. Select **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	<div>Copy to clipboard</div>

Step 4: Create the Console agent

Launch the Console agent directly from the Azure Marketplace.

About this task

Creating the Console agent from the Azure Marketplace sets up a virtual machine with a default configuration. [Learn about the default configuration for the Console agent.](#)

Before you begin

You should have the following:

- An Azure subscription.
- A VNet and subnet in your Azure region of choice.
- Details about a proxy server, if your organization requires a proxy for all outgoing internet traffic:
 - IP address
 - Credentials
 - HTTPS certificate
- An SSH public key, if you want to use that authentication method for the Console agent virtual machine. The other option for the authentication method is to use a password.

[Learn about connecting to a Linux VM in Azure](#)

- If you don't want the Console to automatically create an Azure role for the Console agent, then you'll need to create your own [using the policy on this page](#).

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

Steps

1. Go to the NetApp Console agent VM page in the Azure Marketplace.

[Azure Marketplace page for commercial regions](#)

2. Select **Get it now** and then select **Continue**.
3. From the Azure portal, select **Create** and follow the steps to configure the virtual machine.

Note the following as you configure the VM:

- **VM size:** Choose a VM size that meets CPU and RAM requirements. We recommend Standard_D8s_v3.

- **Disks:** The Console agent can perform optimally with either HDD or SSD disks.
- **Network security group:** The Console agent requires inbound connections using SSH, HTTP, and HTTPS.

[View security group rules for Azure.](#)

- Identity*: Under **Management**, select **Enable system assigned managed identity**.

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

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

Azure deploys the virtual machine with the specified settings. You should see the virtual machine and Console agent software running in about ten minutes.



If the installation fails, you can view logs and a report to help you troubleshoot. [Learn how to troubleshoot installation issues.](#)

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

`https://ipaddress`

6. After you log in, set up the Console agent:
 - a. Specify the the Console organization to associate with the Console agent.
 - b. Enter a name for the system.
 - c. Under **Are you running in a secured environment?** keep restricted mode disabled.

Keep restricted mode disabled to use the Console in standard mode. You should enable restricted mode only if you have a secure environment and want to disconnect this account from the Console backend services. If that's the case, [follow steps to get started with the Console in restricted mode.](#)

- d. Select **Let's start**.

Result

You have now installed the Console agent and set it up with your the Console organization.

If you have Azure Blob storage in the same Azure subscription where you created the Console agent, you'll see an Azure Blob storage system appear on the **Systems** page automatically. [Learn how to manage Azure Blob storage from the Console](#)

Step 5: Provide permissions to the Console agent

Now that you've created the Console agent, you need to provide it with the permissions that you previously set up. Providing the permissions enables the Console agent to manage your data and storage infrastructure in Azure.

Custom role

Go to the Azure portal and assign the Azure custom role to the Console agent virtual machine for one or more subscriptions.

Steps

1. From the Azure Portal, open the **Subscriptions** service and select your subscription.

It's important to assign the role from the **Subscriptions** service because this specifies the scope of the role assignment at the subscription level. The *scope* defines the set of resources that the access applies to. If you specify a scope at a different level (for example, at the virtual machine level), your ability to complete actions from within the NetApp Console will be affected.

[Microsoft Azure documentation: Understand scope for Azure RBAC](#)

2. Select **Access control (IAM) > Add > Add role assignment**.
3. In the **Role** tab, select the **Console Operator** role and select **Next**.



Console Operator is the default name provided in the policy. If you chose a different name for the role, then select that name instead.

4. In the **Members** tab, complete the following steps:
 - a. Assign access to a **Managed identity**.
 - b. Select **Select members**, select the subscription in which the Console agent virtual machine was created, under **Managed identity**, choose **Virtual machine**, and then select the Console agent virtual machine.
 - c. Select **Select**.
 - d. Select **Next**.
 - e. Select **Review + assign**.
 - f. If you want to manage resources in additional Azure subscriptions, switch to that subscription and then repeat these steps.

What's next?

Go to the [NetApp Console](#) to start using the Console agent.

Service principal

Steps

1. Select **Administration > Credentials**.
2. Select **Add Credentials** and follow the steps in the wizard.
 - a. **Credentials Location**: Select **Microsoft Azure > Agent**.
 - b. **Define Credentials**: Enter information about the Microsoft Entra service principal that grants the required permissions:
 - Application (client) ID
 - Directory (tenant) ID
 - Client Secret
 - c. **Marketplace Subscription**: Associate a Marketplace subscription with these credentials by subscribing now or by selecting an existing subscription.

d. **Review:** Confirm the details about the new credentials and select **Add**.

Result

The Console now has the permissions that it needs to perform actions in Azure on your behalf.

Manually install the Console agent in Azure

To manually install the Console agent on your own Linux host, you need to review host requirements, set up your networking, prepare Azure permissions, install the Console agent, and then provide the permissions that you prepared.

Before you begin

- You should have an [understanding of Console agents](#).
- You should review [Console agent limitations](#).

Step 1: Review host requirements

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



The Console agent reserves the UID and GID range of 19000 to 19200. This range is fixed and cannot be modified. If any third-party software on your host is using UIDs or GIDs within this range, the agent installation will fail. NetApp recommends using a host that is free of third-party software to avoid conflicts.

Dedicated host

The Console agent requires a dedicated host. Any architecture is supported if it meets these size requirements:

- CPU: 8 cores or 8 vCPUs
- RAM: 32 GB
- Disk space: 165 GB is recommended for the host, with the following partition requirements:
 - `/opt`: 120 GiB of space must be available

The agent uses `/opt` to install the `/opt/application/netapp` directory and its contents.

- `/var`: 40 GiB of space must be available

The Console agent requires this space in `/var` because Podman or Docker are architected to create the containers within this directory. Specifically, they will create containers in the `/var/lib/containers/storage` directory and `/var/lib/docker` for Docker. External mounts or symlinks do not work for this space.

Azure VM size

An instance type that meets CPU and RAM requirements. NetApp recommends `Standard_D8s_v3`.

Hypervisor

A bare metal or hosted hypervisor that is certified to run a supported operating system is required.

Operating system and container requirements

The Console agent is supported with the following operating systems when using the Console in standard mode or restricted mode. A container orchestration tool is required before you install the agent.

Operating system	Supported OS versions	Supported agent versions	Required container tool	SELinux
Red Hat Enterprise Linux				
	9.6 <ul style="list-style-type: none">English language versions only.The host must be registered with Red Hat Subscription Management. If it's not registered, the host can't access repositories to update required 3rd-party software during agent installation.	4.0.0 or later with the Console in standard mode or restricted mode	Podman version 5.4.0 with podman-compose 1.5.0. View Podman configuration requirements.	Supported in enforcing mode or permissive mode
	9.1 to 9.4 <ul style="list-style-type: none">English language versions only.The host must be registered with Red Hat Subscription Management. If it's not registered, the host can't access repositories to update required 3rd-party software during agent installation.	3.9.50 or later with the Console in standard mode or restricted mode	Podman version 4.9.4 with podman-compose 1.5.0. View Podman configuration requirements.	Supported in enforcing mode or permissive mode

Operating system	Supported OS versions	Supported agent versions	Required container tool	SELinux
	8.6 to 8.10 <ul style="list-style-type: none"> English language versions only. The host must be registered with Red Hat Subscription Management. If it's not registered, the host can't access repositories to update required 3rd-party software during agent installation. 	3.9.50 or later with the Console in standard mode or restricted mode	Podman version 4.6.1 or 4.9.4 with podman-compose 1.0.6. View Podman configuration requirements.	Supported in enforcing mode or permissive mode
Ubuntu				
	24.04 LTS	3.9.45 or later with the NetApp Console in standard mode or restricted mode	Docker Engine 23.06 to 28.0.0.	Not supported
	22.04 LTS	3.9.50 or later	Docker Engine 23.0.6 to 28.0.0.	Not supported

Step 2: Install Podman or Docker Engine

Depending on your operating system, either Podman or Docker Engine is required before installing the agent.

- Podman is required for Red Hat Enterprise Linux 8 and 9.

[View the supported Podman versions.](#)

- Docker Engine is required for Ubuntu.

[View the supported Docker Engine versions.](#)

Example 2. Steps

Podman

Follow these steps to install and configure Podman:

- Enable and start the podman.socket service
- Install python3
- Install the podman-compose package version 1.0.6
- Add podman-compose to the PATH environment variable
- If using Red Hat Enterprise Linux, verify that your Podman version is using Netavark Aardvark DNS instead of CNI



Adjust the aardvark-dns port (default: 53) after installing the agent to avoid DNS port conflicts. Follow the instructions to configure the port.

Steps

1. Remove the podman-docker package if it's installed on the host.

```
dnf remove podman-docker
rm /var/run/docker.sock
```

2. Install Podman.

You can obtain Podman from official Red Hat Enterprise Linux repositories.

- a. For Red Hat Enterprise Linux 9.6:

```
sudo dnf install podman-5:<version>
```

Where <version> is the supported version of Podman that you're installing. [View the supported Podman versions](#).

- b. For Red Hat Enterprise Linux 9.1 to 9.4:

```
sudo dnf install podman-4:<version>
```

Where <version> is the supported version of Podman that you're installing. [View the supported Podman versions](#).

- c. For Red Hat Enterprise Linux 8:

```
sudo dnf install podman-4:<version>
```

Where <version> is the supported version of Podman that you're installing. [View the supported Podman versions](#).

3. Enable and start the podman.socket service.

```
sudo systemctl enable --now podman.socket
```

4. Install python3.

```
sudo dnf install python3
```

5. Install the EPEL repository package if it's not already available on your system.

This step is required because podman-compose is available from the Extra Packages for Enterprise Linux (EPEL) repository.

6. If using Red Hat Enterprise 9:

a. Install the EPEL repository package.

```
sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-9.noarch.rpm
```

a. Install podman-compose package 1.5.0.

```
sudo dnf install podman-compose-1.5.0
```

7. If using Red Hat Enterprise Linux 8:

a. Install the EPEL repository package.

```
sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
```

b. Install podman-compose package 1.0.6.

```
sudo dnf install podman-compose-1.0.6
```



Using the `dnf install` command meets the requirement for adding podman-compose to the PATH environment variable. The installation command adds podman-compose to `/usr/bin`, which is already included in the `secure_path` option on the host.

c. If using Red Hat Enterprise Linux 8, verify that your Podman version is using NetAvark with Aardvark DNS instead of CNI.

i. Check to see if your networkBackend is set to CNI by running the following command:

```
podman info | grep networkBackend
```

- ii. If the `networkBackend` is set to `CNI`, you'll need to change it to `netavark`.
- iii. Install `netavark` and `aardvark-dns` using the following command:

```
dnf install aardvark-dns netavark
```

- iv. Open the `/etc/containers/containers.conf` file and modify the `network_backend` option to use "netavark" instead of "cni".

If `/etc/containers/containers.conf` doesn't exist, make the configuration changes to `/usr/share/containers/containers.conf`.

- v. Restart podman.

```
systemctl restart podman
```

- vi. Confirm `networkBackend` is now changed to "netavark" using the following command:

```
podman info | grep networkBackend
```

Docker Engine

Follow the documentation from Docker to install Docker Engine.

Steps

1. [View installation instructions from Docker](#)

Follow the steps to install a supported Docker Engine version. Do not install the latest version, as it is unsupported by the Console.

2. Verify that Docker is enabled and running.

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

Step 3: Set up networking

Ensure that the network location where you plan to install the Console agent supports the following requirements. Meeting these requirements enables the Console agent to manage resources and processes within your hybrid cloud environment.

Azure region

If you use Cloud Volumes ONTAP, the Console agent 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](#)

Connections to target networks

The Console agent requires a network connection to the location where you're planning to create and manage systems. For example, the network where you plan to create Cloud Volumes ONTAP systems or a storage system in your on-premises environment.

Outbound internet access

The network location where you deploy the Console agent must have an outbound internet connection to contact specific endpoints.

Endpoints contacted from computers when using the web-based NetApp Console

Computers that access the Console from a web browser must have the ability to contact several endpoints. You'll need to use the Console to set up the Console agent and for day-to-day use of the Console.

[Prepare networking for the NetApp console.](#)

Endpoints contacted from the Console agent

The Console agent requires outbound internet access to contact the following endpoints to manage resources and processes within your public cloud environment for day-to-day operations.

The endpoints listed below are all CNAME entries.

Endpoints	Purpose
https://management.azure.com https://login.microsoftonline.com https://blob.core.windows.net https://core.windows.net	To manage resources in Azure public regions.
https://management.chinacloudapi.cn https://login.chinacloudapi.cn https://blob.core.chinacloudapi.cn https://core.chinacloudapi.cn	To manage resources in Azure China regions.
https://mysupport.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support.
https://signin.b2c.netapp.com	To update NetApp Support Site (NSS) credentials or to add new NSS credentials to the NetApp Console.
https://support.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support as well as to receive software updates for Cloud Volumes ONTAP.

Endpoints	Purpose
https://api.bluexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.bluexp.netapp.com https://cdn.auth0.com	To provide features and services within the NetApp Console.
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	To obtain images for Console agent upgrades. <ul style="list-style-type: none"> When you deploy a new agent, the validation check tests connectivity to current endpoints. If you use previous endpoints, the validation check fails. To avoid this failure, skip the validation check. <p>Although the previous endpoints are still supported, NetApp recommends updating your firewall rules to the current endpoints as soon as possible. Learn how to update your endpoint list.</p> <ul style="list-style-type: none"> When you update to the current endpoints in your firewall, your existing agents will continue to work.

Proxy server

NetApp supports both explicit and transparent proxy configurations. If you are using a transparent proxy, you only need to provide the certificate for the proxy server. If you are using an explicit proxy, you'll also need the IP address and credentials.

- IP address
- Credentials
- HTTPS certificate

Ports

There's no incoming traffic to the Console agent, unless you initiate it or if it is used as a proxy to send AutoSupport messages from Cloud Volumes ONTAP to NetApp Support.

- HTTP (80) and HTTPS (443) provide access to the local UI, which you'll use in rare circumstances.
- SSH (22) is only needed if you need to connect to the host for troubleshooting.
- Inbound connections over port 3128 are required if you deploy Cloud Volumes ONTAP systems in a subnet where an outbound internet connection isn't available.

If Cloud Volumes ONTAP systems don't have an outbound internet connection to send AutoSupport messages, the Console automatically configures those systems to use a proxy server that's included with the Console agent. The only requirement is to ensure that the Console agent's security group allows inbound connections over port 3128. You'll need to open this port after you deploy the Console agent.

Enable NTP

If you're planning to use NetApp Data Classification to scan your corporate data sources, you should enable a Network Time Protocol (NTP) service on both the Console agent and the NetApp Data Classification system so that the time is synchronized between the systems. [Learn more about NetApp Data classification](#)

Step 4: Set up Console agent deployment permissions

You need to provide Azure permissions to the Console agent by using one of the following options:

- Option 1: Assign a custom role to the Azure VM using a system-assigned managed identity.
- Option 2: Provide the Console agent with the credentials for an Azure service principal that has the required permissions.

Follow the steps to prepare permissions for the Console agent.

Create a custom role for Console agent deployment

Note that you can create an Azure custom role using the Azure portal, Azure PowerShell, Azure CLI, or REST API. The following steps show how to create the role using the Azure CLI. If you would prefer to use a different method, refer to [Azure documentation](#)

Steps

1. If you're planning to manually install the software on your own host, enable a system-assigned managed identity on the VM so that you can provide the required Azure permissions through a custom role.

[Microsoft Azure documentation: Configure managed identities for Azure resources on a VM using the Azure portal](#)

2. Copy the contents of the [custom role permissions for the Connector](#) and save them in a JSON file.
3. Modify the JSON file by adding Azure subscription IDs to the assignable scope.

You should add the ID for each Azure subscription that you want to use with the NetApp Console.

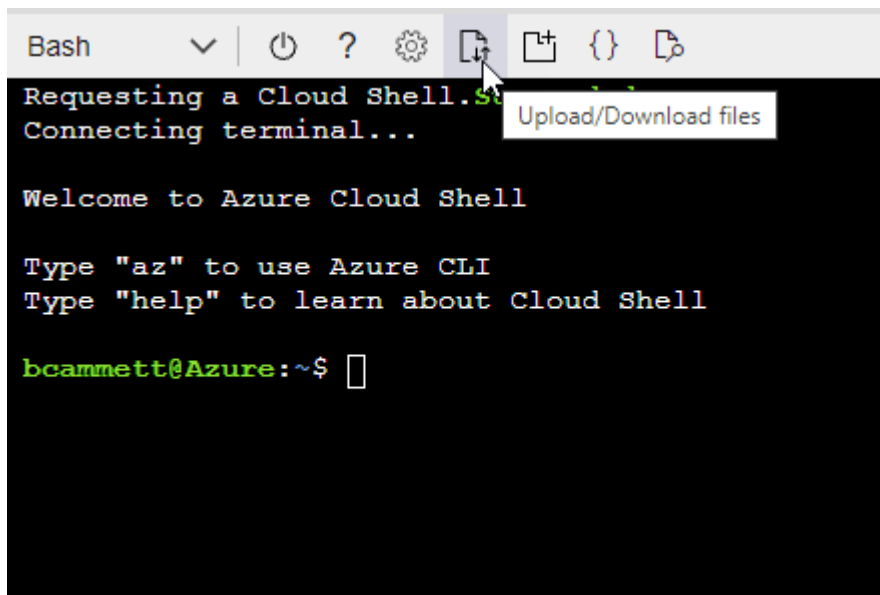
Example

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

4. 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. Use the Azure CLI to create the custom role:

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

Service principal

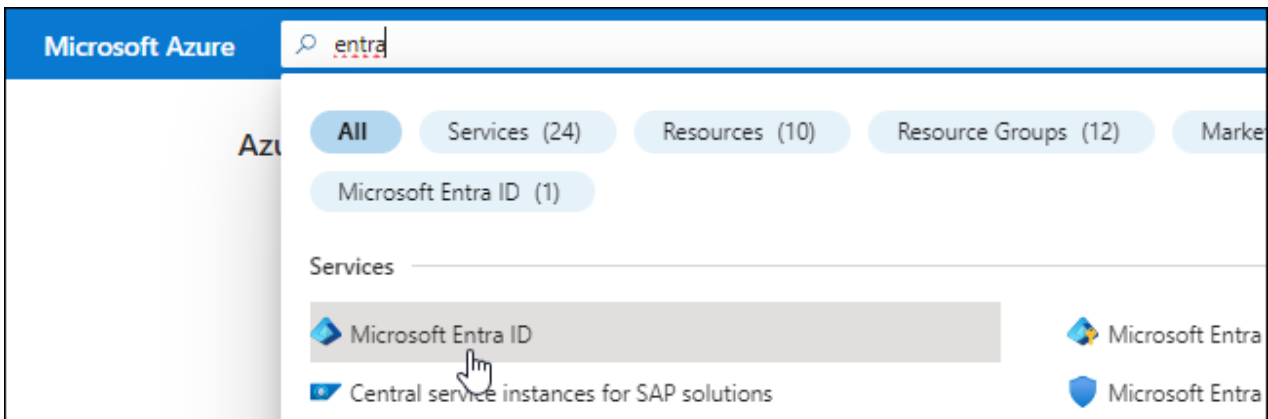
Create and set up a service principal in Microsoft Entra ID and obtain the Azure credentials that the Console agent needs.

Create a Microsoft Entra application for role-based access control

1. Ensure that you have 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](#)

2. From the Azure portal, open the **Microsoft Entra ID** service.



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

You've created the AD application and service principal.

Assign the application to a role

1. Create a custom role:

Note that you can create an Azure custom role using the Azure portal, Azure PowerShell, Azure CLI, or REST API. The following steps show how to create the role using the Azure CLI. If you would prefer to use a different method, refer to [Azure documentation](#)

- a. Copy the contents of the [custom role permissions for the Console agent](#) and save them in a JSON file.

- 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.

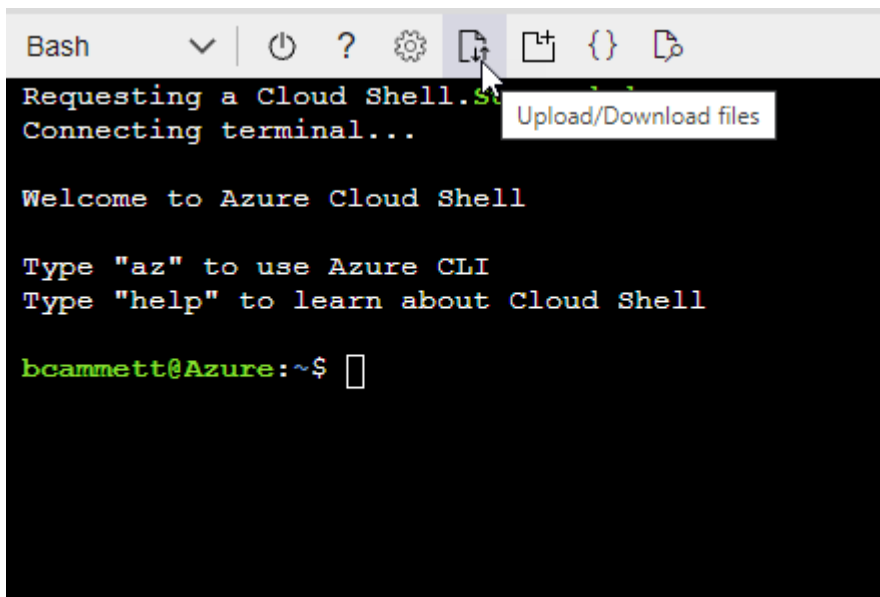
Example

```
"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 steps describe how to create the role by using Bash in Azure Cloud Shell.

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



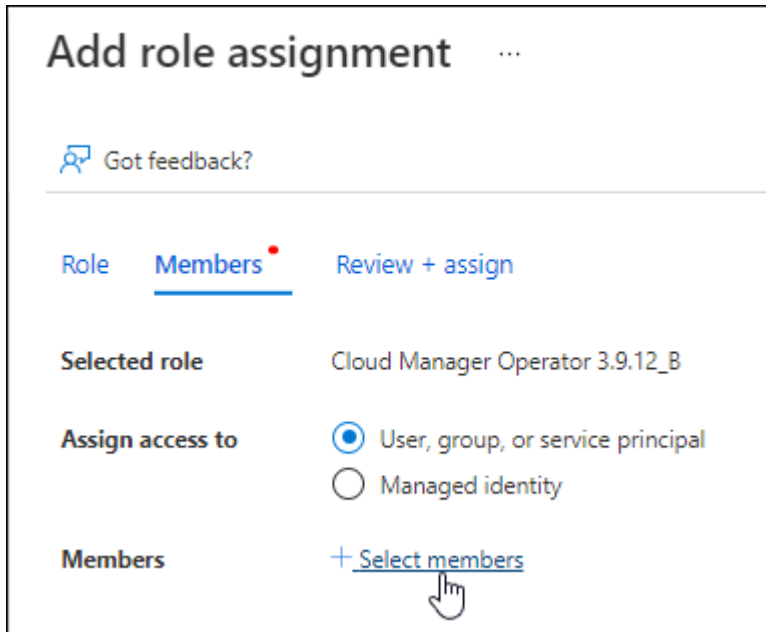
- Use the Azure CLI to create the custom role:

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

You should now have a custom role called Console Operator that you can assign to the Console agent virtual machine.

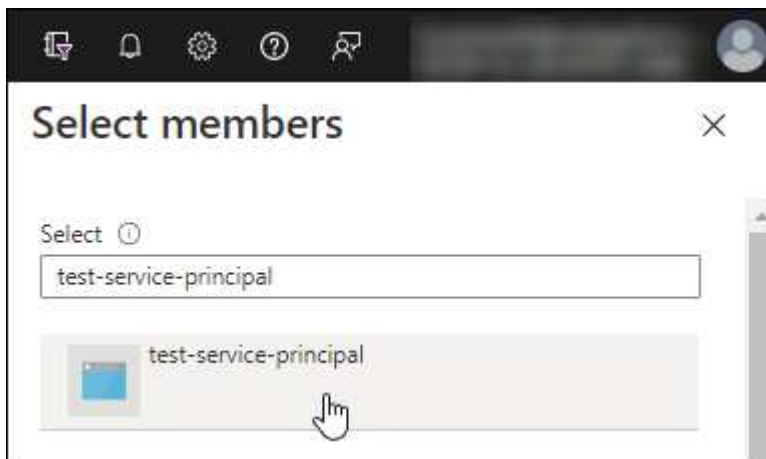
2. Assign the application to the role:
 - a. From the Azure portal, open the **Subscriptions** service.
 - b. Select the subscription.

- c. Select **Access control (IAM) > Add > Add role assignment**.
- d. In the **Role** tab, select the **Console Operator** role and select **Next**.
- e. In the **Members** tab, complete the following steps:
 - Keep **User, group, or service principal** selected.
 - Select **Select members**.



- Search for the name of the application.

Here's an example:



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

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

If you want to deploy Cloud Volumes ONTAP from multiple Azure subscriptions, then you must bind the service principal to each of those subscriptions. In the NetApp Console, you can select

the subscription that you want to use when deploying Cloud Volumes ONTAP.

Add Windows Azure Service Management API permissions

1. In the **Microsoft Entra ID** service, select **App registrations** and select the application.
2. Select **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. Select **Access Azure Service Management as organization users** and then select **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

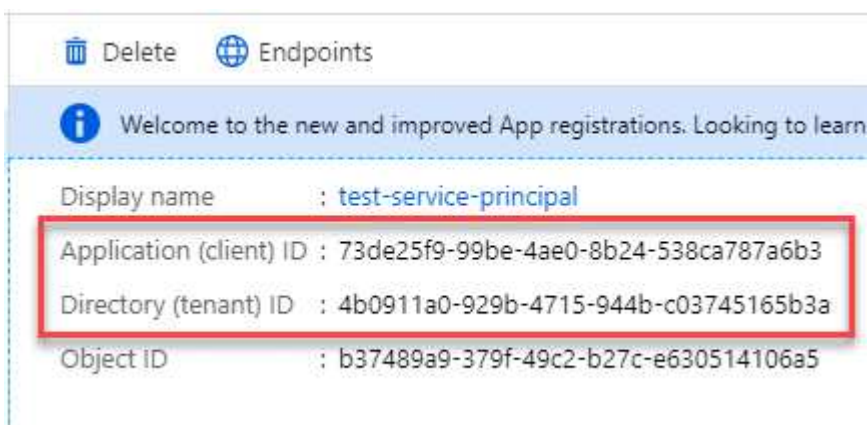


user_impersonation

Access Azure Service Management as organization users (preview) ⓘ

Get the application ID and directory ID for the application

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



When you add the Azure account to the Console, you need to provide the application (client) ID and the directory (tenant) ID for the application. The Console uses the IDs to programmatically sign in.


Create a client secret

1. Open the **Microsoft Entra ID** service.
2. Select **App registrations** and select your application.
3. Select **Certificates & secrets > New client secret**.
4. Provide a description of the secret and a duration.
5. Select **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	Copy to clipboard
test secret	8/16/2020	*sZ1jSe2By:D*-ZRoV4NLfdAcY7:+0vA	

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 the Console when you add an Azure account.

Step 5: Install the Console agent

After the pre-requisites are complete, you can manually install the software on your own Linux host.

Before you begin

You should have the following:

- Root privileges to install the Console agent.
- Details about a proxy server, if a proxy is required for internet access from the Console agent.

You have the option to configure a proxy server after installation but doing so requires restarting the Console agent.

- A CA-signed certificate, if the proxy server uses HTTPS or if the proxy is an intercepting proxy.



You cannot set a certificate for a transparent proxy server when manually installing the Console agent. If you need to set a certificate for a transparent proxy server, you must use the Maintenance Console after installation. Learn more about the [Agent Maintenance Console](#).

- A managed identity enabled on the VM in Azure so that you can provide the required Azure permissions through a custom role.

[Microsoft Azure documentation: Configure managed identities for Azure resources on a VM using the Azure portal](#)

About this task

After installation, the Console agent automatically updates itself if a new version is available.

Steps

1. If the `http_proxy` or `https_proxy` system variables are set on the host, remove them:

```
unset http_proxy
unset https_proxy
```

If you don't remove these system variables, the installation fails.

2. Download the Console agent software and then copy it to the Linux host. You can download it either from the NetApp Console or the NetApp Support site.

- NetApp Console: Go to **Agents > Management > Deploy agent > On-prem > Manual install**.

Choose download the agent installer files or a URL to the files.

- NetApp Support Site (needed if you don't already have access to the Console) [NetApp Support Site](#),

3. Assign permissions to run the script.

```
chmod +x NetApp_Console_Agent_Cloud_<version>
```

Where <version> is the version of the Console agent that you downloaded.

4. If installing in a Government Cloud environment, disable the configuration checks. [Learn how to disable configuration checks for manual installations](#).
5. Run the installation script.

```
./NetApp_Console_Agent_Cloud_<version> --proxy <HTTP or HTTPS proxy server> --cacert <path and file name of a CA-signed certificate>
```

You'll need to add proxy information if your network requires a proxy for internet access. You can add an explicit proxy during installation. The `--proxy` and `--cacert` parameters are optional and you won't be prompted to add them. If you have an explicit proxy server, you will need to enter the parameters as shown.



If you want to configure a transparent proxy, you can do so after you've installed. [Learn about the agent maintenance console](#)

+

Here is an example configuring an explicit proxy server with a CA-signed certificate:

+

```
./NetApp_Console_Agent_Cloud_v4.0.0--proxy  
https://user:password@10.0.0.30:8080/ --cacert /tmp/cacert/certificate.cer
```

+

`--proxy` configures the Console agent to use an HTTP or HTTPS proxy server using one of the following formats:

+

- * `http://address:port`
- * `http://user-name:password@address:port`
- * `http://domain-name%92user-name:password@address:port`
- * `https://address:port`
- * `https://user-name:password@address:port`

* `https://domain-name%92user-name:password@address:port`

+

Note the following:

+

The user can be a local user or domain user.

For a domain user, you must use the ASCII code for a \ as shown above.

The Console agent doesn't support user names or passwords that include the @ character.

If the password includes any of the following special characters, you must escape that special character by prepending it with a backslash: & or !

+

For example:

+

`http://bxpproxyuser:netapp1\!@address:3128`

1. If you used Podman, you'll need to adjust the aardvark-dns port.
 - a. SSH to the Console agent virtual machine.
 - b. Open podman `/usr/share/containers/containers.conf` file and modify the chosen port for Aardvark DNS service. For example, change it to 54.

```
vi /usr/share/containers/containers.conf
```

For example:

```
# Port to use for dns forwarding daemon with netavark in rootful
bridge
# mode and dns enabled.
# Using an alternate port might be useful if other DNS services
should
# run on the machine.
#
dns_bind_port = 54
```

- c. Reboot the Console agent virtual machine.
2. Wait for the installation to complete.

At the end of the installation, the Console agent service (occm) restarts twice if you specified a proxy server.



If the installation fails, you can view the installation report and logs to help you fix the issues.
[Learn how to troubleshoot installation issues.](#)

1. Open a web browser from a host that has a connection to the Console agent virtual machine and enter the following URL:

`https://ipaddress`

2. After you log in, set up the Console agent:

- a. Specify the organization to associate with the Console agent.
- b. Enter a name for the system.
- c. Under **Are you running in a secured environment?** keep restricted mode disabled.

You should keep restricted mode disabled because these steps describe how to use the Console in standard mode. You should enable restricted mode only if you have a secure environment and want to disconnect this account from backend services. If that's the case, [follow steps to get started with the NetApp Console in restricted mode](#).

- d. Select **Let's start**.

If you have Azure Blob storage in the same Azure subscription where you created the Console agent, you'll see an Azure Blob storage system appear on the **Systems** page automatically. [Learn how to manage Azure Blob storage from NetApp Console](#)

Step 6: Provide permissions to NetApp Console

Now that you've installed the Console agent, you need to provide the Console agent with the Azure permissions that you previously set up. Providing the permissions enables the Console to manage your data and storage infrastructure in Azure.

Custom role

Go to the Azure portal and assign the Azure custom role to the Console agent virtual machine for one or more subscriptions.

Steps

1. From the Azure Portal, open the **Subscriptions** service and select your subscription.

It's important to assign the role from the **Subscriptions** service because this specifies the scope of the role assignment at the subscription level. The *scope* defines the set of resources that the access applies to. If you specify a scope at a different level (for example, at the virtual machine level), your ability to complete actions from within the NetApp Console will be affected.

[Microsoft Azure documentation: Understand scope for Azure RBAC](#)

2. Select **Access control (IAM) > Add > Add role assignment**.
3. In the **Role** tab, select the **Console Operator** role and select **Next**.



Console Operator is the default name provided in the policy. If you chose a different name for the role, then select that name instead.

4. In the **Members** tab, complete the following steps:
 - a. Assign access to a **Managed identity**.
 - b. Select **Select members**, select the subscription in which the Console agent virtual machine was created, under **Managed identity**, choose **Virtual machine**, and then select the Console agent virtual machine.
 - c. Select **Select**.
 - d. Select **Next**.
 - e. Select **Review + assign**.
 - f. If you want to manage resources in additional Azure subscriptions, switch to that subscription and then repeat these steps.

What's next?

Go to the [NetApp Console](#) to start using the Console agent.

Service principal

Steps

1. Select **Administration > Credentials**.
2. Select **Add Credentials** and follow the steps in the wizard.
 - a. **Credentials Location**: Select **Microsoft Azure > Agent**.
 - b. **Define Credentials**: Enter information about the Microsoft Entra service principal that grants the required permissions:
 - Application (client) ID
 - Directory (tenant) ID
 - Client Secret
 - c. **Marketplace Subscription**: Associate a Marketplace subscription with these credentials by subscribing now or by selecting an existing subscription.

d. **Review:** Confirm the details about the new credentials and select **Add**.

Result

The Console agent now has the permissions that it needs to perform actions in Azure on your behalf.

Google Cloud

Console agent installation options in Google Cloud

There are a few different ways to create a Console agent in Google Cloud. Directly from the NetApp Console is the most common way.

The following installation options are available:

- [Create the Console agent directly from the Console](#) (this is the standard option)

This action launches a VM instance running Linux and the Console agent software in a VPC of your choice.

- [Create the Console agent using Google Platform](#)

This action also launches a VM instance running Linux and the Console agent software, but the deployment is initiated directly from Google Cloud, rather than from the Console.

- [Download and manually install the software on your own Linux host](#)

The installation option that you choose impacts how you prepare for installation. This includes how you provide the Console with the required permissions that it needs to authenticate and manage resources in Google Cloud.

Create a Console agent in Google Cloud from NetApp Console

You can create a Console agent in Google Cloud from the Console. You need to set up your networking, prepare Google Cloud permissions, enable Google Cloud APIs, and then create the Console agent.

Before you begin

- You should have an [understanding of Console agents](#).
- You should review [Console agent limitations](#).

Step 1: Set up networking

Set up networking to ensure the Console agent can manage resources, with connections to target networks and outbound internet access.

VPC and subnet

When you create the Console agent, you need to specify the VPC and subnet where it should reside.

Connections to target networks

The Console agent requires a network connection to the location where you're planning to create and manage systems. For example, the network where you plan to create Cloud Volumes ONTAP systems or a storage system in your on-premises environment.

Outbound internet access

The network location where you deploy the Console agent must have an outbound internet connection to contact specific endpoints.

Endpoints contacted from the Console agent

The Console agent requires outbound internet access to contact the following endpoints to manage resources and processes within your public cloud environment for day-to-day operations.

The endpoints listed below are all CNAME entries.

Endpoints	Purpose
https://www.googleapis.com/compute/v1/ https://compute.googleapis.com/compute/v1 https://cloudresourcemanager.googleapis.com/v1/projects https://www.googleapis.com/compute/beta https://storage.googleapis.com/storage/v1 https://www.googleapis.com/storage/v1 https://iam.googleapis.com/v1 https://cloudkms.googleapis.com/v1 https://config.googleapis.com/v1/projects	To manage resources in Google Cloud.
https://mysupport.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support.
https://signin.b2c.netapp.com	To update NetApp Support Site (NSS) credentials or to add new NSS credentials to the NetApp Console.
https://support.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support as well as to receive software updates for Cloud Volumes ONTAP.
https://api.bluelxp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.bluelxp.netapp.com https://cdn.auth0.com	To provide features and services within the NetApp Console.

Endpoints	Purpose
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	<p>To obtain images for Console agent upgrades.</p> <ul style="list-style-type: none"> When you deploy a new agent, the validation check tests connectivity to current endpoints. If you use previous endpoints, the validation check fails. To avoid this failure, skip the validation check. <p>Although the previous endpoints are still supported, NetApp recommends updating your firewall rules to the current endpoints as soon as possible. Learn how to update your endpoint list.</p> <ul style="list-style-type: none"> When you update to the current endpoints in your firewall, your existing agents will continue to work.

Endpoints contacted from the NetApp console

As you use the web-based NetApp Console that's provided through the SaaS layer, it contacts several endpoints to complete data management tasks. This includes endpoints that are contacted to deploy the Console agent from the the Console.

[View the list of endpoints contacted from the NetApp console.](#)

Proxy server

NetApp supports both explicit and transparent proxy configurations. If you are using a transparent proxy, you only need to provide the certificate for the proxy server. If you are using an explicit proxy, you'll also need the IP address and credentials.

- IP address
- Credentials
- HTTPS certificate

Ports

There's no incoming traffic to the Console agent, unless you initiate it or if it is used as a proxy to send AutoSupport messages from Cloud Volumes ONTAP to NetApp Support.

- HTTP (80) and HTTPS (443) provide access to the local UI, which you'll use in rare circumstances.
- SSH (22) is only needed if you need to connect to the host for troubleshooting.
- Inbound connections over port 3128 are required if you deploy Cloud Volumes ONTAP systems in a subnet where an outbound internet connection isn't available.

If Cloud Volumes ONTAP systems don't have an outbound internet connection to send AutoSupport messages, the Console automatically configures those systems to use a proxy server that's included with the Console agent. The only requirement is to ensure that the Console agent's security group allows inbound connections over port 3128. You'll need to open this port after you deploy the Console agent.

Enable NTP

If you're planning to use NetApp Data Classification to scan your corporate data sources, you should enable a Network Time Protocol (NTP) service on both the Console agent and the NetApp Data Classification system so that the time is synchronized between the systems. [Learn more about NetApp Data classification](#)

Implement this networking requirement after creating the Console agent.

Step 2: Set up permissions to create the Console agent

Before you can deploy a Console agent from the Console, you need to set up permissions for the Google Platform user who deploys the Console agent VM.

Steps

1. Create a custom role in Google Platform:
 - a. Create a YAML file that includes the following permissions:

```
title: Console agent deployment policy
description: Permissions for the user who deploys the Console agent
stage: GA
includedPermissions:

- cloudbuild.builds.get
- compute.disks.create
- compute.disks.get
- compute.disks.list
- compute.disks.setLabels
- compute.disks.use
- compute.firewalls.create
- compute.firewalls.delete
- compute.firewalls.get
- compute.firewalls.list
- compute.globalOperations.get
- compute.images.get
- compute.images.getFromFamily
- compute.images.list
- compute.images.useReadOnly
- compute.instances.attachDisk
- compute.instances.create
- compute.instances.get
- compute.instances.list
- compute.instances.setDeletionProtection
- compute.instances.setLabels
- compute.instances.setMachineType
- compute.instances.setMetadata
- compute.instances.setTags
- compute.instances.start
- compute.instances.updateDisplayDevice
```


- compute.machineTypes.get
- compute.networks.get
- compute.networks.list
- compute.networks.updatePolicy
- compute.projects.get
- compute.regions.get
- compute.regions.list
- compute.subnetworks.get
- compute.subnetworks.list
- compute.zoneOperations.get
- compute.zones.get
- compute.zones.list
- config.deployments.create
- config.deployments.get
- config.deployments.delete
- config.deployments.deleteState
- config.deployments.get
- config.deployments.getState
- config.deployments.list
- config.deployments.update
- config.deployments.updateState
- config.previews.get
- config.previews.list
- config.revisions.get
- config.resources.list
- deploymentmanager.compositeTypes.get
- deploymentmanager.compositeTypes.list
- deploymentmanager.deployments.create
- deploymentmanager.deployments.delete
- deploymentmanager.deployments.get
- deploymentmanager.deployments.list
- deploymentmanager.manifests.get
- deploymentmanager.manifests.list
- deploymentmanager.operations.get
- deploymentmanager.operations.list
- deploymentmanager.resources.get
- deploymentmanager.resources.list
- deploymentmanager.typeProviders.get
- deploymentmanager.typeProviders.list
- deploymentmanager.types.get
- deploymentmanager.types.list
- resourcemanager.projects.get
- compute.instances.setServiceAccount
- iam.serviceAccounts.actAs
- iam.serviceAccounts.create
- iam.serviceAccounts.list

- `iam.serviceAccountKeys.create`
- `storage.buckets.create`
- `storage.buckets.get`
- `storage.objects.create`
- `storage.folders.create`
- `storage.objects.list`

- From Google Cloud, activate cloud shell.
- Upload the YAML file that includes the required permissions.
- Create a custom role by using the `gcloud iam roles create` command.

The following example creates a role named "agentDeployment" at the project level:

```
gcloud iam roles create connectorDeployment --project=myproject --file=agent-deployment.yaml
```

[Google Cloud docs: Creating and managing custom roles](#)

- Assign this custom role to the user who will deploy the Console agent from the Console or by using `gcloud`.

[Google Cloud docs: Grant a single role](#)

Step 3: Create a Google Cloud service account to use with the agent

A Google Cloud service account is required to provide the Console agent with the permissions that the Console needs to manage resources in Google Cloud. When you create the Console agent, you'll need to associate this service account with the Console agent VM.

It's your responsibility to update the custom role as new permissions are added in subsequent releases. If new permissions are required, they will be listed in the release notes.

Steps

- Create a custom role in Google Cloud:
 - Create a YAML file that includes the contents of the [service account permissions for the Console agent](#).
 - From Google Cloud, activate cloud shell.
 - Upload the YAML file that includes the required permissions.
 - Create a custom role by using the `gcloud iam roles create` command.

The following example creates a role named "agent" at the project level:

```
gcloud iam roles create connector --project=myproject --file=agent.yaml
```

[Google Cloud docs: Creating and managing custom roles](#)

- Create a service account in Google Cloud and assign the role to the service account:
 - From the IAM & Admin service, select **Service Accounts > Create Service Account**.
 - Enter service account details and select **Create and Continue**.
 - Select the role that you just created.

d. Finish the remaining steps to create the role.

[Google Cloud docs: Creating a service account](#)

3. If you plan to deploy Cloud Volumes ONTAP systems in different projects than the project where the Console agent resides, then you'll need to provide the Console agent's service account with access to those projects.

For example, let's say the Console agent is in project 1 and you want to create Cloud Volumes ONTAP systems in project 2. You'll need to grant access to the service account in project 2.

- a. From the IAM & Admin service, select the Google Cloud project where you want to create Cloud Volumes ONTAP systems.
- b. On the **IAM** page, select **Grant Access** and provide the required details.
 - Enter the email of the Console agent's service account.
 - Select the Console agent's custom role.
 - Select **Save**.

For more details, refer to [Google Cloud documentation](#)

Step 4: Set up shared VPC permissions

If you are using a shared VPC to deploy resources into a service project, then you'll need to prepare your permissions.

This table is for reference and your environment should reflect the permissions table when IAM configuration is complete.

View shared VPC permissions

Identity	Creator	Hosted in	Service project permissions	Host project permissions	Purpose
Google account to deploy the agent	Custom	Service Project	Agent deployment policy	compute.network User	Deploying the agent in the service project
agent service account	Custom	Service project	Agent service account policy	compute.network User deploymentmanager.editor	Deploying and maintaining Cloud Volumes ONTAP and services in the service project
Cloud Volumes ONTAP service account	Custom	Service project	storage.admin member: NetApp Console service account as serviceAccount.user	N/A	(Optional) For NetApp Cloud Tiering and NetApp Backup and Recovery
Google APIs service agent	Google Cloud	Service project	(Default) Editor	compute.network User	Interacts with Google Cloud APIs on behalf of deployment. Allows the Console to use the shared network.
Google Compute Engine default service account	Google Cloud	Service project	(Default) Editor	compute.network User	Deploys Google Cloud instances and compute infrastructure on behalf of deployment. Allows the Console 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 the Console create them for you. The NetApp Console creates a deployment in the host project which contains the VPC0 firewall rule if no rule is specified.
2. firewall.create and firewall.delete are only required if you are not passing firewall rules to the deployment and are choosing to let the Console create them for you. These permissions reside in the Console account .yaml file. If you are deploying an HA pair using a shared VPC, these permissions will be used to create the firewall rules for VPC1, 2 and 3. For all other deployments, these permissions will also be used to create rules for VPC0.
3. For Cloud 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.

Step 5: Enable Google Cloud APIs

You must enable several Google Cloud APIs before deploying the Console agent and Cloud Volumes ONTAP.

Step

1. Enable the following Google Cloud APIs in your project:

- Cloud Infrastructure Manager API
- Cloud Deployment Manager V2 API
- Cloud Logging API
- Cloud Resource Manager API
- Compute Engine API
- Identity and Access Management (IAM) API
- Cloud Key Management Service (KMS) API

(Required only if you are planning to use NetApp Backup and Recovery with customer-managed encryption keys (CMEK))

[Google Cloud documentation: Enabling APIs](#)

Step 6: Create the Console agent

Create a Console agent directly from the Console.

Creating the Console agent deploys a virtual machine instance in Google Cloud using a default configuration. Do not switch to a smaller VM instance with fewer CPUs or less RAM after creating the Console agent. [Learn about the default configuration for the Console agent.](#)



When you deploy an agent in Google Cloud, the agent creates a bucket to store deployment files.

Before you begin

You should have the following:

- The required Google Cloud permissions to create the Console agent and a service account for the Console agent VM.
- A VPC and subnet that meets networking requirements.
- Details about a proxy server, if a proxy is required for internet access from the Console agent.

Steps

1. Select **Administration > Agents**.
2. On the **Overview** page, select **Deploy agent > Google Cloud**
3. On the **Deploying an agent** page, review the details about what you'll need. You have two options:
 - a. Select **Continue** to prepare for deployment by using the in-product guide. Each step in the in-product guide includes the information that's contained on this page of the documentation.
 - b. Select **Skip to Deployment** if you already prepared by following the steps on this page.
4. Follow the steps in the wizard to create the Console agent:
 - 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.

- **Details:** 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.
- **Network tags:** Add a network tag to the Console agent instance if using a transparent proxy. Network tags must start with a lowercase letter and can contain lowercase letters, numbers, and hyphens. Tags must end with a lowercase letter or number. For example, you might use the tag "console-agent-proxy".
- **Firewall Policy:** Choose whether to create a new firewall policy or whether to select an existing firewall policy that allows the required inbound and outbound rules.

Firewall rules in Google Cloud

5. Review your selections to verify that your set up is correct.

- The **Validate agent configuration** check box is marked by default to have the Console validate the network connectivity requirements when you deploy. If the Console fails to deploy the agent, it provides a report to help you troubleshoot. If the deployment succeeds, no report is provided.

If you are still using the [previous endpoints](#) used for agent upgrades, the validation fails with an error. To avoid this, unmark the check box to skip the validation check.

6. Select **Add**.

The agent is ready in approximately 10 minutes; stay on the page until the process completes.

Result

After the process completes, the Console agent is available for use.



If the deployment fails, you can download a report and logs from the Console to help you fix the issues. [Learn how to troubleshoot installation issues.](#)

If you have Google Cloud Storage buckets in the same Google Cloud account where you created the Console agent, you'll see a Google Cloud Storage system appear on the **Systems** page automatically. [Learn how to manage Google Cloud Storage from the Console](#)

Create a Console agent from Google Cloud

To create a Console agent in Google Cloud by using Google Cloud, you need to set up your networking, prepare Google Cloud permissions, enable Google Cloud APIs, and then create the Console agent.

Before you begin

- You should have a [understanding of Console agents](#).
- You should review [Console agent limitations](#).

Step 1: Set up networking

Set up networking to enable the Console agent to manage resources and connect to target networks and the internet.

VPC and subnet

When you create the Console agent, you need to specify the VPC and subnet where it should reside.

Connections to target networks

The Console agent requires a network connection to the location where you're planning to create and manage systems. For example, the network where you plan to create Cloud Volumes ONTAP systems or a storage system in your on-premises environment.

Outbound internet access

The network location where you deploy the Console agent must have an outbound internet connection to contact specific endpoints.

Endpoints contacted from the Console agent

The Console agent requires outbound internet access to contact the following endpoints to manage resources and processes within your public cloud environment for day-to-day operations.

The endpoints listed below are all CNAME entries.

Endpoints	Purpose
https://www.googleapis.com/compute/v1/ https://compute.googleapis.com/compute/v1 https://cloudresourcemanager.googleapis.com/v1/projects https://www.googleapis.com/compute/beta https://storage.googleapis.com/storage/v1 https://www.googleapis.com/storage/v1 https://iam.googleapis.com/v1 https://cloudkms.googleapis.com/v1 https://config.googleapis.com/v1/projects	To manage resources in Google Cloud.
https://mysupport.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support.
https://signin.b2c.netapp.com	To update NetApp Support Site (NSS) credentials or to add new NSS credentials to the NetApp Console.
https://support.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support as well as to receive software updates for Cloud Volumes ONTAP.
https://api.bluexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.bluexp.netapp.com https://cdn.auth0.com	To provide features and services within the NetApp Console.

Endpoints	Purpose
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	<p>To obtain images for Console agent upgrades.</p> <ul style="list-style-type: none"> When you deploy a new agent, the validation check tests connectivity to current endpoints. If you use previous endpoints, the validation check fails. To avoid this failure, skip the validation check. <p>Although the previous endpoints are still supported, NetApp recommends updating your firewall rules to the current endpoints as soon as possible. Learn how to update your endpoint list.</p> <ul style="list-style-type: none"> When you update to the current endpoints in your firewall, your existing agents will continue to work.

Endpoints contacted from the NetApp console

As you use the web-based NetApp Console that's provided through the SaaS layer, it contacts several endpoints to complete data management tasks. This includes endpoints that are contacted to deploy the Console agent from the the Console.

[View the list of endpoints contacted from the NetApp console.](#)

Proxy server

NetApp supports both explicit and transparent proxy configurations. If you are using a transparent proxy, you only need to provide the certificate for the proxy server. If you are using an explicit proxy, you'll also need the IP address and credentials.

- IP address
- Credentials
- HTTPS certificate

Ports

There's no incoming traffic to the Console agent, unless you initiate it or if it is used as a proxy to send AutoSupport messages from Cloud Volumes ONTAP to NetApp Support.

- HTTP (80) and HTTPS (443) provide access to the local UI, which you'll use in rare circumstances.
- SSH (22) is only needed if you need to connect to the host for troubleshooting.
- Inbound connections over port 3128 are required if you deploy Cloud Volumes ONTAP systems in a subnet where an outbound internet connection isn't available.

If Cloud Volumes ONTAP systems don't have an outbound internet connection to send AutoSupport messages, the Console automatically configures those systems to use a proxy server that's included with the Console agent. The only requirement is to ensure that the Console agent's security group allows inbound connections over port 3128. You'll need to open this port after you deploy the Console agent.

Enable NTP

If you're planning to use NetApp Data Classification to scan your corporate data sources, you should enable a Network Time Protocol (NTP) service on both the Console agent and the NetApp Data Classification system so that the time is synchronized between the systems. [Learn more about NetApp Data classification](#)

Implement this networking requirement after creating the Console agent.

Step 2: Set up permissions to create the Console agent

Set up permissions for the Google Cloud user to deploy the Console agent VM from Google Cloud.

Steps

1. Create a custom role in Google Platform:
 - a. Create a YAML file that includes the following permissions:

```
title: Console agent deployment policy
description: Permissions for the user who deploys the Console
agent
stage: GA
includedPermissions:

- cloudbuild.builds.get
- compute.disks.create
- compute.disks.get
- compute.disks.list
- compute.disks.setLabels
- compute.disks.use
- compute.firewalls.create
- compute.firewalls.delete
- compute.firewalls.get
- compute.firewalls.list
- compute.globalOperations.get
- compute.images.get
- compute.images.getFromFamily
- compute.images.list
- compute.images.useReadOnly
- compute.instances.attachDisk
- compute.instances.create
- compute.instances.get
- compute.instances.list
- compute.instances.setDeletionProtection
- compute.instances.setLabels
- compute.instances.setMachineType
- compute.instances.setMetadata
- compute.instances.setTags
- compute.instances.start
- compute.instances.updateDisplayDevice
- compute.machineTypes.get
- compute.networks.get
- compute.networks.list
- compute.networks.updatePolicy
- compute.projects.get
- compute.regions.get
- compute.regions.list
- compute.subnetworks.get
- compute.subnetworks.list
- compute.zoneOperations.get
- compute.zones.get
- compute.zones.list
- config.deployments.create
```

- config.operations.get
- config.deployments.delete
- config.deployments.deleteState
- config.deployments.get
- config.deployments.getState
- config.deployments.list
- config.deployments.update
- config.deployments.updateState
- config.preview.get
- config.preview.list
- config.revisions.get
- config.resources.list
- deploymentmanager.compositeTypes.get
- deploymentmanager.compositeTypes.list
- deploymentmanager.deployments.create
- deploymentmanager.deployments.delete
- deploymentmanager.deployments.get
- deploymentmanager.deployments.list
- deploymentmanager.manifests.get
- deploymentmanager.manifests.list
- deploymentmanager.operations.get
- deploymentmanager.operations.list
- deploymentmanager.resources.get
- deploymentmanager.resources.list
- deploymentmanager.typeProviders.get
- deploymentmanager.typeProviders.list
- deploymentmanager.types.get
- deploymentmanager.types.list
- resourcemanager.projects.get
- compute.instances.setServiceAccount
- iam.serviceAccounts.actAs
- iam.serviceAccounts.create
- iam.serviceAccounts.list
- iam.serviceAccountKeys.create
- storage.buckets.create
- storage.buckets.get
- storage.objects.create
- storage.folders.create
- storage.objects.list

- b. From Google Cloud, activate cloud shell.
- c. Upload the YAML file that includes the required permissions.
- d. Create a custom role by using the `gcloud iam roles create` command.

The following example creates a role named "connectorDeployment" at the project level:

```
gcloud iam roles create connectorDeployment --project=myproject --file=connector-deployment.yaml
```

[Google Cloud docs: Creating and managing custom roles](#)

2. Assign this custom role to the user who deploys the Console agent from Google Cloud.

[Google Cloud docs: Grant a single role](#)

Step 3: Set up permissions for the Console agent operations

A Google Cloud service account is required to provide the Console agent with the permissions that the Console needs to manage resources in Google Cloud. When you create the Console agent, you'll need to associate this service account with the Console agent VM.

It's your responsibility to update the custom role as new permissions are added in subsequent releases. If new permissions are required, they will be listed in the release notes.

Steps

1. Create a custom role in Google Cloud:
 - a. Create a YAML file that includes the contents of the [service account permissions for the Console agent](#).
 - b. From Google Cloud, activate cloud shell.
 - c. Upload the YAML file that includes the required permissions.
 - d. Create a custom role by using the `gcloud iam roles create` command.

The following example creates a role named "agent" at the project level:

```
gcloud iam roles create connector --project=myproject --file=agent.yaml
```

[Google Cloud docs: Creating and managing custom roles](#)

2. Create a service account in Google Cloud and assign the role to the service account:
 - a. From the IAM & Admin service, select **Service Accounts > Create Service Account**.
 - b. Enter service account details and select **Create and Continue**.
 - c. Select the role that you just created.
 - d. Finish the remaining steps to create the role.

[Google Cloud docs: Creating a service account](#)

3. If you plan to deploy Cloud Volumes ONTAP systems in different projects than the project where the Console agent resides, then you'll need to provide the Console agent's service account with access to those projects.

For example, let's say the Console agent is in project 1 and you want to create Cloud Volumes ONTAP systems in project 2. You'll need to grant access to the service account in project 2.

- a. From the IAM & Admin service, select the Google Cloud project where you want to create Cloud Volumes ONTAP systems.
- b. On the **IAM** page, select **Grant Access** and provide the required details.
 - Enter the email of the Console agent's service account.

- Select the Console agent's custom role.
- Select **Save**.

For more details, refer to [Google Cloud documentation](#)

Step 4: Set up shared VPC permissions

If you are using a shared VPC to deploy resources into a service project, then you'll need to prepare your permissions.

This table is for reference and your environment should reflect the permissions table when IAM configuration is complete.

View shared VPC permissions

Identity	Creator	Hosted in	Service project permissions	Host project permissions	Purpose
Google account to deploy the agent	Custom	Service Project	Agent deployment policy	compute.network User	Deploying the agent in the service project
agent service account	Custom	Service project	Agent service account policy	compute.network User deploymentmanager.editor	Deploying and maintaining Cloud Volumes ONTAP and services in the service project
Cloud Volumes ONTAP service account	Custom	Service project	storage.admin member: NetApp Console service account as serviceAccount.user	N/A	(Optional) For NetApp Cloud Tiering and NetApp Backup and Recovery
Google APIs service agent	Google Cloud	Service project	(Default) Editor	compute.network User	Interacts with Google Cloud APIs on behalf of deployment. Allows the Console to use the shared network.
Google Compute Engine default service account	Google Cloud	Service project	(Default) Editor	compute.network User	Deploys Google Cloud instances and compute infrastructure on behalf of deployment. Allows the Console 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 the Console create them for you. The NetApp Console creates a deployment in the host project which contains the VPC0 firewall rule if no rule is specified.
2. firewall.create and firewall.delete are only required if you are not passing firewall rules to the deployment and are choosing to let the Console create them for you. These permissions reside in the Console account .yaml file. If you are deploying an HA pair using a shared VPC, these permissions will be used to create the firewall rules for VPC1, 2 and 3. For all other deployments, these permissions will also be used to create rules for VPC0.
3. For Cloud 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.

Step 5: Enable Google Cloud APIs

Enable several Google Cloud APIs before deploying the Console agent and Cloud Volumes ONTAP.

Step

1. Enable the following Google Cloud APIs in your project:

- Cloud Infrastructure Manager API
- Cloud Deployment Manager V2 API
- Cloud Logging API
- Cloud Resource Manager API
- Compute Engine API
- Identity and Access Management (IAM) API
- Cloud Key Management Service (KMS) API

(Required only if you are planning to use NetApp Backup and Recovery with customer-managed encryption keys (CMEK))

[Google Cloud documentation: Enabling APIs](#)

Step 6: Create the Console agent

Create a Console agent by using Google Cloud.

Creating the Console agent deploys a VM instance in Google Cloud with the default configuration. Do not switch to a smaller VM instance with fewer CPUs or less RAM after you create the Console agent. [Learn about the default configuration for the Console agent.](#)

Before you begin

You should have the following:

- The required Google Cloud permissions to create the Console agent and a service account for the Console agent VM.
- A VPC and subnet that meets networking requirements.
- An understanding of VM instance requirements.
 - **CPU:** 8 cores or 8 vCPUs
 - **RAM:** 32 GB
 - **Machine type:** We recommend n2-standard-8.

The Console agent is supported in Google Cloud on a VM instance with an OS that supports Shielded VM features.

Steps

1. Log in to the Google Cloud SDK using your preferred method.

This example uses a local shell with the gcloud SDK installed, but you can also use the Google Cloud Shell.

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=n2-standard-8
  --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 Console agent instance

network-path

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

subnet-path

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

kms-key-path

(Optional) Add a KMS key to encrypt the Console agent'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 Console agent. The Console agent instance and software should be running in approximately five minutes.

4. Open a web browser and enter the Console agent host URL:

The Console host URL can be a localhost, a private IP address, or a public IP address, depending on the configuration of the host. For example, if the Console agent 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 Console agent host.

5. After you log in, set up the Console agent:

- a. Specify the Console organization to associate with the Console agent.

[Learn about identity and access management.](#)

- b. Enter a name for the system.

Result

The Console agent is now installed and set up with your Console organization.

Open a web browser and go to the [NetApp Console](#) to start using the Console agent.

Manually install the Console agent in Google Cloud

To manually install the Console agent on your own Linux host, you need to review host requirements, set up your networking, prepare Google Cloud permissions, enable Google Cloud APIs, install the Console, and then provide the permissions that you prepared.

Before you begin

- You should have an [understanding of Console agents](#).
- You should review [Console agent limitations](#).

Step 1: Review host requirements

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



The Console agent reserves the UID and GID range of 19000 to 19200. This range is fixed and cannot be modified. If any third-party software on your host is using UIDs or GIDs within this range, the agent installation will fail. NetApp recommends using a host that is free of third-party software to avoid conflicts.

Dedicated host

The Console agent requires a dedicated host. Any architecture is supported if it meets these size requirements:

- CPU: 8 cores or 8 vCPUs
- RAM: 32 GB
- Disk space: 165 GB is recommended for the host, with the following partition requirements:
 - `/opt`: 120 GiB of space must be available

The agent uses `/opt` to install the `/opt/application/netapp` directory and its contents.

- `/var`: 40 GiB of space must be available

The Console agent requires this space in `/var` because Podman or Docker are architected to create the containers within this directory. Specifically, they will create containers in the `/var/lib/containers/storage` directory and `/var/lib/docker` for Docker. External mounts or symlinks do not work for this space.

Google Cloud machine type

An instance type that meets CPU and RAM requirements. NetApp recommends n2-standard-8.

The Console agent is supported in Google Cloud on a VM instance with an OS that supports [Shielded VM features](#)

Hypervisor

A bare metal or hosted hypervisor that is certified to run a supported operating system is required.

Operating system and container requirements

The Console agent is supported with the following operating systems when using the Console in standard mode or restricted mode. A container orchestration tool is required before you install the agent.

Operating system	Supported OS versions	Supported agent versions	Required container tool	SELinux
Red Hat Enterprise Linux				

Operating system	Supported OS versions	Supported agent versions	Required container tool	SELinux
	9.6 <ul style="list-style-type: none"> English language versions only. The host must be registered with Red Hat Subscription Management. If it's not registered, the host can't access repositories to update required 3rd-party software during agent installation. 	4.0.0 or later with the Console in standard mode or restricted mode	Podman version 5.4.0 with podman-compose 1.5.0. View Podman configuration requirements.	Supported in enforcing mode or permissive mode
	9.1 to 9.4 <ul style="list-style-type: none"> English language versions only. The host must be registered with Red Hat Subscription Management. If it's not registered, the host can't access repositories to update required 3rd-party software during agent installation. 	3.9.50 or later with the Console in standard mode or restricted mode	Podman version 4.9.4 with podman-compose 1.5.0. View Podman configuration requirements.	Supported in enforcing mode or permissive mode

Operating system	Supported OS versions	Supported agent versions	Required container tool	SELinux
	8.6 to 8.10 <ul style="list-style-type: none"> English language versions only. The host must be registered with Red Hat Subscription Management. If it's not registered, the host can't access repositories to update required 3rd-party software during agent installation. 	3.9.50 or later with the Console in standard mode or restricted mode	Podman version 4.6.1 or 4.9.4 with podman-compose 1.0.6. View Podman configuration requirements.	Supported in enforcing mode or permissive mode
Ubuntu				
	24.04 LTS	3.9.45 or later with the NetApp Console in standard mode or restricted mode	Docker Engine 23.06 to 28.0.0.	Not supported
	22.04 LTS	3.9.50 or later	Docker Engine 23.0.6 to 28.0.0.	Not supported

Google Cloud machine type

An instance type that meets CPU and RAM requirements. NetApp recommends n2-standard-8.

The Console agent is supported in Google Cloud on a VM instance with an OS that supports [Shielded VM features](#)

Step 2: Install Podman or Docker Engine

Depending on your operating system, either Podman or Docker Engine is required before installing the agent.

- Podman is required for Red Hat Enterprise Linux 8 and 9.

[View the supported Podman versions.](#)

- Docker Engine is required for Ubuntu.

[View the supported Docker Engine versions.](#)

Example 3. Steps

Podman

Follow these steps to install and configure Podman:

- Enable and start the podman.socket service
- Install python3
- Install the podman-compose package version 1.0.6
- Add podman-compose to the PATH environment variable
- If using Red Hat Enterprise Linux, verify that your Podman version is using Netavark Aardvark DNS instead of CNL



Adjust the aardvark-dns port (default: 53) after installing the agent to avoid DNS port conflicts. Follow the instructions to configure the port.

Steps

1. Remove the podman-docker package if it's installed on the host.

```
dnf remove podman-docker
rm /var/run/docker.sock
```

2. Install Podman.

You can obtain Podman from official Red Hat Enterprise Linux repositories.

- a. For Red Hat Enterprise Linux 9.6:

```
sudo dnf install podman-5:<version>
```

Where <version> is the supported version of Podman that you're installing. [View the supported Podman versions](#).

- b. For Red Hat Enterprise Linux 9.1 to 9.4:

```
sudo dnf install podman-4:<version>
```

Where <version> is the supported version of Podman that you're installing. [View the supported Podman versions](#).

- c. For Red Hat Enterprise Linux 8:

```
sudo dnf install podman-4:<version>
```

Where <version> is the supported version of Podman that you're installing. [View the supported Podman versions](#).

3. Enable and start the podman.socket service.

```
sudo systemctl enable --now podman.socket
```

4. Install python3.

```
sudo dnf install python3
```

5. Install the EPEL repository package if it's not already available on your system.

This step is required because podman-compose is available from the Extra Packages for Enterprise Linux (EPEL) repository.

6. If using Red Hat Enterprise 9:

a. Install the EPEL repository package.

```
sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-9.noarch.rpm
```

a. Install podman-compose package 1.5.0.

```
sudo dnf install podman-compose-1.5.0
```

7. If using Red Hat Enterprise Linux 8:

a. Install the EPEL repository package.

```
sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
```

b. Install podman-compose package 1.0.6.

```
sudo dnf install podman-compose-1.0.6
```



Using the `dnf install` command meets the requirement for adding podman-compose to the PATH environment variable. The installation command adds podman-compose to `/usr/bin`, which is already included in the `secure_path` option on the host.

c. If using Red Hat Enterprise Linux 8, verify that your Podman version is using NetAvark with Aardvark DNS instead of CNI.

i. Check to see if your networkBackend is set to CNI by running the following command:

```
podman info | grep networkBackend
```

- ii. If the `networkBackend` is set to `CNI`, you'll need to change it to `netavark`.
- iii. Install `netavark` and `aardvark-dns` using the following command:

```
dnf install aardvark-dns netavark
```

- iv. Open the `/etc/containers/containers.conf` file and modify the `network_backend` option to use "netavark" instead of "cni".

If `/etc/containers/containers.conf` doesn't exist, make the configuration changes to `/usr/share/containers/containers.conf`.

- v. Restart podman.

```
systemctl restart podman
```

- vi. Confirm `networkBackend` is now changed to "netavark" using the following command:

```
podman info | grep networkBackend
```

Docker Engine

Follow the documentation from Docker to install Docker Engine.

Steps

1. [View installation instructions from Docker](#)

Follow the steps to install a supported Docker Engine version. Do not install the latest version, as it is unsupported by the Console.

2. Verify that Docker is enabled and running.

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

Step 3: Set up networking

Set up your networking so the Console agent can manage resources and processes within your hybrid cloud environment. For example, you need to ensure that connections are available to target networks and that outbound internet access is available.

Connections to target networks

The Console agent requires a network connection to the location where you're planning to create and

manage systems. For example, the network where you plan to create Cloud Volumes ONTAP systems or a storage system in your on-premises environment.

Outbound internet access

The network location where you deploy the Console agent must have an outbound internet connection to contact specific endpoints.

Endpoints contacted from computers when using the web-based NetApp Console

Computers that access the Console from a web browser must have the ability to contact several endpoints. You'll need to use the Console to set up the Console agent and for day-to-day use of the Console.

[Prepare networking for the NetApp console.](#)

Endpoints contacted from the Console agent

The Console agent requires outbound internet access to contact the following endpoints to manage resources and processes within your public cloud environment for day-to-day operations.

The endpoints listed below are all CNAME entries.

Endpoints	Purpose
https://www.googleapis.com/compute/v1/ https://compute.googleapis.com/compute/v1 https://cloudresourcemanager.googleapis.com/v1/projects https://www.googleapis.com/compute/beta https://storage.googleapis.com/storage/v1 https://www.googleapis.com/storage/v1 https://iam.googleapis.com/v1 https://cloudkms.googleapis.com/v1 https://config.googleapis.com/v1/projects	To manage resources in Google Cloud.
https://mysupport.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support.
https://signin.b2c.netapp.com	To update NetApp Support Site (NSS) credentials or to add new NSS credentials to the NetApp Console.
https://support.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support as well as to receive software updates for Cloud Volumes ONTAP.
https://api.bluexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.bluexp.netapp.com https://cdn.auth0.com	To provide features and services within the NetApp Console.

Endpoints	Purpose
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	<p>To obtain images for Console agent upgrades.</p> <ul style="list-style-type: none"> When you deploy a new agent, the validation check tests connectivity to current endpoints. If you use previous endpoints, the validation check fails. To avoid this failure, skip the validation check. <p>Although the previous endpoints are still supported, NetApp recommends updating your firewall rules to the current endpoints as soon as possible. Learn how to update your endpoint list.</p> <ul style="list-style-type: none"> When you update to the current endpoints in your firewall, your existing agents will continue to work.

Proxy server

NetApp supports both explicit and transparent proxy configurations. If you are using a transparent proxy, you only need to provide the certificate for the proxy server. If you are using an explicit proxy, you'll also need the IP address and credentials.

- IP address
- Credentials
- HTTPS certificate

Ports

There's no incoming traffic to the Console agent, unless you initiate it or if it is used as a proxy to send AutoSupport messages from Cloud Volumes ONTAP to NetApp Support.

- HTTP (80) and HTTPS (443) provide access to the local UI, which you'll use in rare circumstances.
- SSH (22) is only needed if you need to connect to the host for troubleshooting.
- Inbound connections over port 3128 are required if you deploy Cloud Volumes ONTAP systems in a subnet where an outbound internet connection isn't available.

If Cloud Volumes ONTAP systems don't have an outbound internet connection to send AutoSupport messages, the Console automatically configures those systems to use a proxy server that's included with the Console agent. The only requirement is to ensure that the Console agent's security group allows inbound connections over port 3128. You'll need to open this port after you deploy the Console agent.

Enable NTP

If you're planning to use NetApp Data Classification to scan your corporate data sources, you should enable a Network Time Protocol (NTP) service on both the Console agent and the NetApp Data Classification system so that the time is synchronized between the systems. [Learn more about NetApp Data classification](#)

Step 4: Set up permissions for the Console agent

A Google Cloud service account is required to provide the Console agent with the permissions that the

Console needs to manage resources in Google Cloud. When you create the Console agent, you'll need to associate this service account with the Console agent VM.

It's your responsibility to update the custom role as new permissions are added in subsequent releases. If new permissions are required, they will be listed in the release notes.

Steps

1. Create a custom role in Google Cloud:
 - a. Create a YAML file that includes the contents of the [service account permissions for the Console agent](#).
 - b. From Google Cloud, activate cloud shell.
 - c. Upload the YAML file that includes the required permissions.
 - d. Create a custom role by using the `gcloud iam roles create` command.

The following example creates a role named "agent" at the project level:

```
gcloud iam roles create connector --project=myproject --file=agent.yaml
```

[Google Cloud docs: Creating and managing custom roles](#)

2. Create a service account in Google Cloud and assign the role to the service account:
 - a. From the IAM & Admin service, select **Service Accounts > Create Service Account**.
 - b. Enter service account details and select **Create and Continue**.
 - c. Select the role that you just created.
 - d. Finish the remaining steps to create the role.

[Google Cloud docs: Creating a service account](#)

3. If you plan to deploy Cloud Volumes ONTAP systems in different projects than the project where the Console agent resides, then you'll need to provide the Console agent's service account with access to those projects.

For example, let's say the Console agent is in project 1 and you want to create Cloud Volumes ONTAP systems in project 2. You'll need to grant access to the service account in project 2.

- a. From the IAM & Admin service, select the Google Cloud project where you want to create Cloud Volumes ONTAP systems.
- b. On the **IAM** page, select **Grant Access** and provide the required details.
 - Enter the email of the Console agent's service account.
 - Select the Console agent's custom role.
 - Select **Save**.

For more details, refer to [Google Cloud documentation](#)

Step 5: Set up shared VPC permissions

If you are using a shared VPC to deploy resources into a service project, then you'll need to prepare your permissions.

This table is for reference and your environment should reflect the permissions table when IAM configuration is

complete.

View shared VPC permissions

Identity	Creator	Hosted in	Service project permissions	Host project permissions	Purpose
Google account to deploy the agent	Custom	Service Project	Agent deployment policy	compute.network User	Deploying the agent in the service project
agent service account	Custom	Service project	Agent service account policy	compute.network User deploymentmanager.editor	Deploying and maintaining Cloud Volumes ONTAP and services in the service project
Cloud Volumes ONTAP service account	Custom	Service project	storage.admin member: NetApp Console service account as serviceAccount.user	N/A	(Optional) For NetApp Cloud Tiering and NetApp Backup and Recovery
Google APIs service agent	Google Cloud	Service project	(Default) Editor	compute.network User	Interacts with Google Cloud APIs on behalf of deployment. Allows the Console to use the shared network.
Google Compute Engine default service account	Google Cloud	Service project	(Default) Editor	compute.network User	Deploys Google Cloud instances and compute infrastructure on behalf of deployment. Allows the Console 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 the Console create them for you. The NetApp Console creates a deployment in the host project which contains the VPC0 firewall rule if no rule is specified.
2. firewall.create and firewall.delete are only required if you are not passing firewall rules to the deployment and are choosing to let the Console create them for you. These permissions reside in the Console account .yaml file. If you are deploying an HA pair using a shared VPC, these permissions will be used to create the firewall rules for VPC1, 2 and 3. For all other deployments, these permissions will also be used to create rules for VPC0.
3. For Cloud 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.

Step 6: Enable Google Cloud APIs

Several Google Cloud APIs must be enabled before you can deploy a Console agent in Google Cloud.

Step

1. Enable the following Google Cloud APIs in your project:

- Cloud Infrastructure Manager API
- Cloud Deployment Manager V2 API
- Cloud Logging API
- Cloud Resource Manager API
- Compute Engine API
- Identity and Access Management (IAM) API
- Cloud Key Management Service (KMS) API

(Required only if you are planning to use NetApp Backup and Recovery with customer-managed encryption keys (CMEK))

[Google Cloud documentation: Enabling APIs](#)

Step 7: Install the Console agent

After the pre-requisites are complete, you can manually install the software on your own Linux host.

When you deploy an agent, the system also creates a Google Cloud bucket to store deployment files.

Before you begin

You should have the following:

- Root privileges to install the Console agent.
- Details about a proxy server, if a proxy is required for internet access from the Console agent.

You have the option to configure a proxy server after installation but doing so requires restarting the Console agent.

- A CA-signed certificate, if the proxy server uses HTTPS or if the proxy is an intercepting proxy.



You cannot set a certificate for a transparent proxy server when manually installing the Console agent. If you need to set a certificate for a transparent proxy server, you must use the Maintenance Console after installation. Learn more about the [Agent Maintenance Console](#).

About this task

After installation, the Console agent automatically updates itself if a new version is available.

Steps

1. If the `http_proxy` or `https_proxy` system variables are set on the host, remove them:

```
unset http_proxy
unset https_proxy
```

If you don't remove these system variables, the installation fails.

2. Download the Console agent software and then copy it to the Linux host. You can download it either from the NetApp Console or the NetApp Support site.

- NetApp Console: Go to **Agents > Management > Deploy agent > On-prem > Manual install**.

Choose download the agent installer files or a URL to the files.

- NetApp Support Site (needed if you don't already have access to the Console) [NetApp Support Site](#),

3. Assign permissions to run the script.

```
chmod +x NetApp_Console_Agent_Cloud_<version>
```

Where <version> is the version of the Console agent that you downloaded.

4. If installing in a Government Cloud environment, disable the configuration checks. [Learn how to disable configuration checks for manual installations](#).
5. Run the installation script.

```
./NetApp_Console_Agent_Cloud_<version> --proxy <HTTP or HTTPS proxy server> --cacert <path and file name of a CA-signed certificate>
```

You'll need to add proxy information if your network requires a proxy for internet access. You can add an explicit proxy during installation. The `--proxy` and `--cacert` parameters are optional and you won't be prompted to add them. If you have an explicit proxy server, you will need to enter the parameters as shown.



If you want to configure a transparent proxy, you can do so after you've installed. [Learn about the agent maintenance console](#)

+

Here is an example configuring an explicit proxy server with a CA-signed certificate:

+

```
./NetApp_Console_Agent_Cloud_v4.0.0--proxy
https://user:password@10.0.0.30:8080/ --cacert /tmp/cacert/certificate.cer
```

+

`--proxy` configures the Console agent to use an HTTP or HTTPS proxy server using one of the following formats:

+
* http://address:port
* http://user-name:password@address:port
* http://domain-name%92user-name:password@address:port
* https://address:port
* https://user-name:password@address:port
* https://domain-name%92user-name:password@address:port

+
Note the following:

+
The user can be a local user or domain user.
For a domain user, you must use the ASCII code for a \ as shown above.
The Console agent doesn't support user names or passwords that include the @ character.
If the password includes any of the following special characters, you must escape that special character by prepending it with a backslash: & or !

+
For example:

+
`http://bxpproxyuser:netapp1!@address:3128`

1. If you used Podman, you'll need to adjust the aardvark-dns port.
 - a. SSH to the Console agent virtual machine.
 - b. Open podman `/usr/share/containers/containers.conf` file and modify the chosen port for Aardvark DNS service. For example, change it to 54.

```
vi /usr/share/containers/containers.conf
```

For example:

```
# Port to use for dns forwarding daemon with netavark in rootful
bridge
# mode and dns enabled.
# Using an alternate port might be useful if other DNS services
should
# run on the machine.
#
dns_bind_port = 54
```

- c. Reboot the Console agent virtual machine.
2. Wait for the installation to complete.

At the end of the installation, the Console agent service (occm) restarts twice if you specified a proxy server.



If the installation fails, you can view the installation report and logs to help you fix the issues. [Learn how to troubleshoot installation issues.](#)

1. Open a web browser from a host that has a connection to the Console agent virtual machine and enter the following URL:

`https://ipaddress`

2. After you log in, set up the Console agent:
 - a. Specify the organization to associate with the Console agent.
 - b. Enter a name for the system.
 - c. Under **Are you running in a secured environment?** keep restricted mode disabled.

You should keep restricted mode disabled because these steps describe how to use the Console in standard mode. You should enable restricted mode only if you have a secure environment and want to disconnect this account from backend services. If that's the case, [follow steps to get started with the NetApp Console in restricted mode.](#)

- d. Select **Let's start.**



If the installation fails, you can view logs and a report to help you troubleshoot. [Learn how to troubleshoot installation issues.](#)

If you have Google Cloud Storage buckets in the same Google Cloud account where you created the Console agent, you'll see a Google Cloud Storage system appear on the **Systems** page automatically. [Learn how to manage Google Cloud Storage from the NetApp Console](#)

Step 8: Provide permissions to Console agent

You need to provide the Console agent with the Google Cloud permissions that you previously set up. Providing the permissions enables the Console agent to manage your data and storage infrastructure in Google Cloud.

Steps

1. Go to the Google Cloud portal and assign the service account to the Console agent VM instance.

[Google Cloud documentation: Changing the service account and access scopes for an instance](#)

2. If you want to manage resources in other Google Cloud projects, grant access by adding the service account with the Console agent role to that project. You'll need to repeat this step for each project.

Install an agent on-premises

Manually install a Console agent on-premises

Install a Console agent on-premises and then log in and set it up to work with your Console organization.



If you are a VMWare user, you can use an OVA to install a Console agent in your VCenter. [Learn more about installing an agent in a VCenter.](#)

Before you install, you'll need to ensure your host (VM or Linux host) meets requirements and ensure that the Console agent will have outbound access to the internet as well as targeted networks. If you plan to NetApp data services, or cloud storage options such as Cloud Volumes ONTAP, you'll need to create credentials in your cloud provider to add to the Console so that the Console agent can perform actions in the cloud on your behalf.

Prepare to install the Console agent

Before you install a Console agent, you should ensure you have a host machine that meets installation requirements. You'll also need to work with your network administrator to ensure that the Console agent has outbound access to required endpoints and connections to targeted networks.

Review Console agent host requirements

Run the Console agent on a x86 host that meets operating system, RAM, and port requirements. Ensure that your host meets these requirements before you install the Console agent.



The Console agent reserves the UID and GID range of 19000 to 19200. This range is fixed and cannot be modified. If any third-party software on your host is using UIDs or GIDs within this range, the agent installation will fail. NetApp recommends using a host that is free of third-party software to avoid conflicts.

Dedicated host

The Console agent requires a dedicated host. Any architecture is supported if it meets these size requirements:

- CPU: 8 cores or 8 vCPUs
- RAM: 32 GB
- Disk space: 165 GB is recommended for the host, with the following partition requirements:
 - `/opt`: 120 GiB of space must be available

The agent uses `/opt` to install the `/opt/application/netapp` directory and its contents.

- `/var`: 40 GiB of space must be available

The Console agent requires this space in `/var` because Podman or Docker are architected to create the containers within this directory. Specifically, they will create containers in the `/var/lib/containers/storage` directory and `/var/lib/docker` for Docker. External mounts or symlinks do not work for this space.

Hypervisor

A bare metal or hosted hypervisor that is certified to run a supported operating system is required.

Operating system and container requirements

The Console agent is supported with the following operating systems when using the Console in standard mode or restricted mode. A container orchestration tool is required before you install the agent.

Operating system	Supported OS versions	Supported agent versions	Required container tool	SELinux
Red Hat Enterprise Linux				
	9.6 <ul style="list-style-type: none"> English language versions only. The host must be registered with Red Hat Subscription Management. If it's not registered, the host can't access repositories to update required 3rd-party software during agent installation. 	4.0.0 or later with the Console in standard mode or restricted mode	Podman version 5.4.0 with podman-compose 1.5.0. View Podman configuration requirements.	Supported in enforcing mode or permissive mode
	9.1 to 9.4 <ul style="list-style-type: none"> English language versions only. The host must be registered with Red Hat Subscription Management. If it's not registered, the host can't access repositories to update required 3rd-party software during agent installation. 	3.9.50 or later with the Console in standard mode or restricted mode	Podman version 4.9.4 with podman-compose 1.5.0. View Podman configuration requirements.	Supported in enforcing mode or permissive mode

Operating system	Supported OS versions	Supported agent versions	Required container tool	SELinux
	8.6 to 8.10 <ul style="list-style-type: none"> English language versions only. The host must be registered with Red Hat Subscription Management. If it's not registered, the host can't access repositories to update required 3rd-party software during agent installation. 	3.9.50 or later with the Console in standard mode or restricted mode	Podman version 4.6.1 or 4.9.4 with podman-compose 1.0.6. View Podman configuration requirements.	Supported in enforcing mode or permissive mode
Ubuntu				
	24.04 LTS	3.9.45 or later with the NetApp Console in standard mode or restricted mode	Docker Engine 23.06 to 28.0.0.	Not supported
	22.04 LTS	3.9.50 or later	Docker Engine 23.0.6 to 28.0.0.	Not supported

Set up network access for the Console agent

Set up network access to ensure the Console agent can manage resources. It needs connections to target networks and outbound internet access to specific endpoints.

Connections to target networks

The Console agent requires a network connection to the location where you're planning to create and manage systems. For example, the network where you plan to create Cloud Volumes ONTAP systems or a storage system in your on-premises environment.

Outbound internet access

The network location where you deploy the Console agent must have an outbound internet connection to contact specific endpoints.

Endpoints contacted from computers when using the web-based NetApp Console

Computers that access the Console from a web browser must have the ability to contact several endpoints. You'll need to use the Console to set up the Console agent and for day-to-day use of the Console.

[Prepare networking for the NetApp console.](#)

Endpoints contacted from the Console agent

The Console agent requires outbound internet access to contact the following endpoints to manage resources and processes within your public cloud environment for day-to-day operations.

The endpoints listed below are all CNAME entries.



A Console agent installed on your premises cannot manage resources in Google Cloud. If you want to manage Google Cloud resources, you need to install an agent in Google Cloud.

AWS

When the Console agent is installed on-premises, it needs network access to the following AWS endpoints in order to manage NetApp systems (such as Cloud Volumes ONTAP) deployed in AWS.

Endpoints contacted from the Console agent

The Console agent requires outbound internet access to contact the following endpoints to manage resources and processes within your public cloud environment for day-to-day operations.

The endpoints listed below are all CNAME entries.

Endpoints	Purpose
AWS services (amazonaws.com): <ul style="list-style-type: none">• CloudFormation• Elastic Compute Cloud (EC2)• Identity and Access Management (IAM)• Key Management Service (KMS)• Security Token Service (STS)• Simple Storage Service (S3)	To manage AWS resources. The endpoint depends on your AWS region. Refer to AWS documentation for details
Amazon FsX for NetApp ONTAP: <ul style="list-style-type: none">• api.workloads.netapp.com	The web-based console contacts this endpoint to interact with the Workload Factory APIs to manage and operate FSx for ONTAP based workloads.
https://mysupport.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support.
https://signin.b2c.netapp.com	To update NetApp Support Site (NSS) credentials or to add new NSS credentials to the NetApp Console.
https://support.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support as well as to receive software updates for Cloud Volumes ONTAP.
https://api.blueexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.blueexp.netapp.com https://cdn.auth0.com	To provide features and services within the NetApp Console.

Endpoints	Purpose
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	<p>To obtain images for Console agent upgrades.</p> <ul style="list-style-type: none"> When you deploy a new agent, the validation check tests connectivity to current endpoints. If you use previous endpoints, the validation check fails. To avoid this failure, skip the validation check. <p>Although the previous endpoints are still supported, NetApp recommends updating your firewall rules to the current endpoints as soon as possible. Learn how to update your endpoint list.</p> <ul style="list-style-type: none"> When you update to the current endpoints in your firewall, your existing agents will continue to work.

Azure

When the Console agent is installed on-premises, it needs network access to the following Azure endpoints in order to manage NetApp systems (such as Cloud Volumes ONTAP) deployed in Azure.

Endpoints	Purpose
https://management.azure.com https://login.microsoftonline.com https://blob.core.windows.net https://core.windows.net	To manage resources in Azure public regions.
https://management.chinacloudapi.cn https://login.chinacloudapi.cn https://blob.core.chinacloudapi.cn https://core.chinacloudapi.cn	To manage resources in Azure China regions.
https://mysupport.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support.
https://signin.b2c.netapp.com	To update NetApp Support Site (NSS) credentials or to add new NSS credentials to the NetApp Console.
https://support.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support as well as to receive software updates for Cloud Volumes ONTAP.

Endpoints	Purpose
https://api.bluexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.bluexp.netapp.com https://cdn.auth0.com	To provide features and services within the NetApp Console.
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	To obtain images for Console agent upgrades. <ul style="list-style-type: none"> When you deploy a new agent, the validation check tests connectivity to current endpoints. If you use previous endpoints, the validation check fails. To avoid this failure, skip the validation check. <p>Although the previous endpoints are still supported, NetApp recommends updating your firewall rules to the current endpoints as soon as possible. Learn how to update your endpoint list.</p> <ul style="list-style-type: none"> When you update to the current endpoints in your firewall, your existing agents will continue to work.

Proxy server

NetApp supports both explicit and transparent proxy configurations. If you are using a transparent proxy, you only need to provide the certificate for the proxy server. If you are using an explicit proxy, you'll also need the IP address and credentials.

- IP address
- Credentials
- HTTPS certificate

Ports

There's no incoming traffic to the Console agent, unless you initiate it or if it is used as a proxy to send AutoSupport messages from Cloud Volumes ONTAP to NetApp Support.

- HTTP (80) and HTTPS (443) provide access to the local UI, which you'll use in rare circumstances.
- SSH (22) is only needed if you need to connect to the host for troubleshooting.
- Inbound connections over port 3128 are required if you deploy Cloud Volumes ONTAP systems in a subnet where an outbound internet connection isn't available.

If Cloud Volumes ONTAP systems don't have an outbound internet connection to send AutoSupport messages, the Console automatically configures those systems to use a proxy server that's included with the Console agent. The only requirement is to ensure that the Console agent's security group allows inbound connections over port 3128. You'll need to open this port after you deploy the Console agent.

Enable NTP

If you're planning to use NetApp Data Classification to scan your corporate data sources, you should enable a Network Time Protocol (NTP) service on both the Console agent and the NetApp Data Classification system so that the time is synchronized between the systems. [Learn more about NetApp Data classification](#)

Create Console agent cloud permissions for AWS or Azure

If you want to use NetApp data services in AWS or Azure with an on-premises Console agent, then you need to set up permissions in your cloud provider and then you add the credentials to the Console agent after you install it.



You must install the Console agent in Google Cloud to manage any resources that reside there.

AWS

When the Console agent is installed on-premises, you need to provide the Console with AWS permissions by adding access keys for an IAM user who has the required permissions.

You must use this authentication method if the Console agent is installed on-premises. You can't use an IAM role.

Steps

1. Log in to the AWS console and navigate to the IAM service.
2. Create a policy:
 - a. Select **Policies > Create policy**.
 - b. Select **JSON** and copy and paste the contents of the [IAM policy for the Console agent](#).
 - c. Finish the remaining steps to create the policy.

Depending on the NetApp data services that you're planning to use, you might need to create a second policy.

For standard regions, the permissions are spread across two policies. Two policies are required due to a maximum character size limit for managed policies in AWS. [Learn more about IAM policies for the Console agent](#).

3. Attach the policies to an IAM user.
 - [AWS Documentation: Creating IAM Roles](#)
 - [AWS Documentation: Adding and Removing IAM Policies](#)
4. Ensure that the user has an access key that you can add to the NetApp Console after you install the Console agent.

Result

You should now have access keys for an IAM user who has the required permissions. After you install the Console agent, associate these credentials with the Console agent from the Console.

Azure

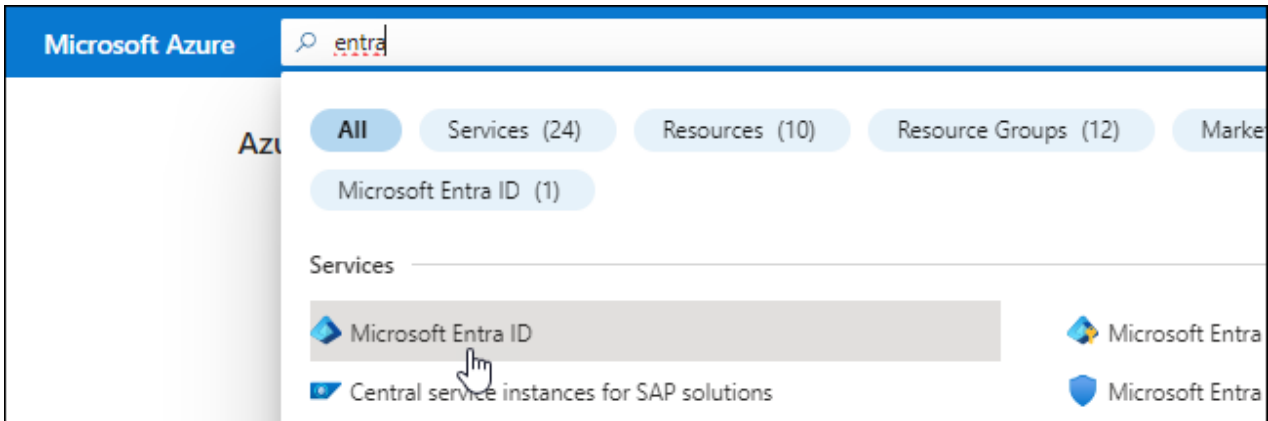
When the Console agent is installed on-premises, you need to provide the Console agent with Azure permissions by setting up a service principal in Microsoft Entra ID and obtaining the Azure credentials that the Console agent needs.

Create a Microsoft Entra application for role-based access control

1. Ensure that you have 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](#)

2. From the Azure portal, open the **Microsoft Entra ID** service.



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

You've created the AD application and service principal.

Assign the application to a role

1. Create a custom role:

Note that you can create an Azure custom role using the Azure portal, Azure PowerShell, Azure CLI, or REST API. The following steps show how to create the role using the Azure CLI. If you would prefer to use a different method, refer to [Azure documentation](#)

- a. Copy the contents of the [custom role permissions for the Console agent](#) and save them in a JSON file.
- 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.

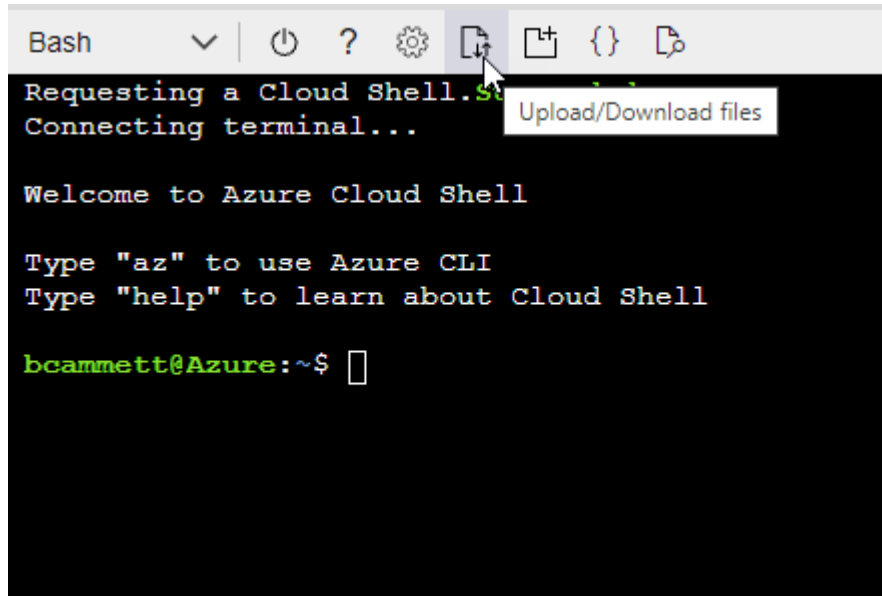
Example

```
"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 steps describe how to create the role by using Bash in Azure Cloud Shell.

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



- Use the Azure CLI to create the custom role:

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

You should now have a custom role called Console Operator that you can assign to the Console agent virtual machine.

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

Add role assignment ...

Got feedback?

Role **Members** Review + assign

Selected role Cloud Manager Operator 3.9.12_B

Assign access to ☒ User, group, or service principal
☐ Managed identity

Members + [Select members](#)

- Search for the name of the application.

Here's an example:

Select members ✕

Select ⓘ

test-service-principal

test-service-principal

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

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

If you want to deploy Cloud Volumes ONTAP from multiple Azure subscriptions, then you must bind the service principal to each of those subscriptions. In the NetApp Console, you can select the subscription that you want to use when deploying Cloud Volumes ONTAP.

Add Windows Azure Service Management API permissions

1. In the **Microsoft Entra ID** service, select **App registrations** and select the application.
2. Select **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. Select **Access Azure Service Management as organization users** and then select **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

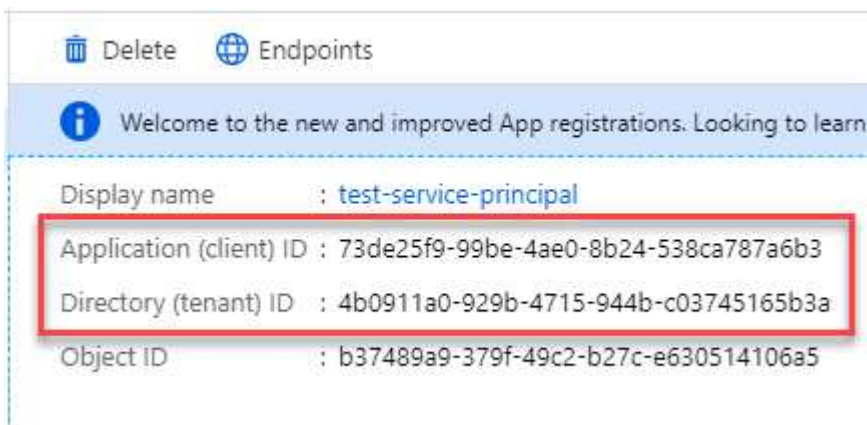


user_impersonation

Access Azure Service Management as organization users (preview) ⓘ

Get the application ID and directory ID for the application

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



When you add the Azure account to the Console, you need to provide the application (client) ID and the directory (tenant) ID for the application. The Console uses the IDs to programmatically sign in.


Create a client secret

1. Open the **Microsoft Entra ID** service.
2. Select **App registrations** and select your application.
3. Select **Certificates & secrets > New client secret**.
4. Provide a description of the secret and a duration.
5. Select **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	Copy to clipboard
test secret	8/16/2020	*sZ1jSe2By:D*-ZRoV4NLfdAcY7:+0vA	

Manually install a Console agent

When you manually install a Console agent, you need to prepare your machine environment so that it meets requirements. You'll need an Linux machine and you'll need to install Podman or Docker, depending on your Linux operating system.

Install Podman or Docker Engine

Depending on your operating system, either Podman or Docker Engine is required before installing the agent.

- Podman is required for Red Hat Enterprise Linux 8 and 9.

[View the supported Podman versions.](#)

- Docker Engine is required for Ubuntu.

[View the supported Docker Engine versions.](#)

Example 4. Steps

Podman

Follow these steps to install and configure Podman:

- Enable and start the podman.socket service
- Install python3
- Install the podman-compose package version 1.0.6
- Add podman-compose to the PATH environment variable
- If using Red Hat Enterprise Linux, verify that your Podman version is using Netavark Aardvark DNS instead of CNL



Adjust the aardvark-dns port (default: 53) after installing the agent to avoid DNS port conflicts. Follow the instructions to configure the port.

Steps

1. Remove the podman-docker package if it's installed on the host.

```
dnf remove podman-docker
rm /var/run/docker.sock
```

2. Install Podman.

You can obtain Podman from official Red Hat Enterprise Linux repositories.

- a. For Red Hat Enterprise Linux 9.6:

```
sudo dnf install podman-5:<version>
```

Where <version> is the supported version of Podman that you're installing. [View the supported Podman versions](#).

- b. For Red Hat Enterprise Linux 9.1 to 9.4:

```
sudo dnf install podman-4:<version>
```

Where <version> is the supported version of Podman that you're installing. [View the supported Podman versions](#).

- c. For Red Hat Enterprise Linux 8:

```
sudo dnf install podman-4:<version>
```

Where <version> is the supported version of Podman that you're installing. [View the supported Podman versions](#).

3. Enable and start the podman.socket service.

```
sudo systemctl enable --now podman.socket
```

4. Install python3.

```
sudo dnf install python3
```

5. Install the EPEL repository package if it's not already available on your system.

This step is required because podman-compose is available from the Extra Packages for Enterprise Linux (EPEL) repository.

6. If using Red Hat Enterprise 9:

a. Install the EPEL repository package.

```
sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-9.noarch.rpm
```

a. Install podman-compose package 1.5.0.

```
sudo dnf install podman-compose-1.5.0
```

7. If using Red Hat Enterprise Linux 8:

a. Install the EPEL repository package.

```
sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
```

b. Install podman-compose package 1.0.6.

```
sudo dnf install podman-compose-1.0.6
```



Using the `dnf install` command meets the requirement for adding podman-compose to the PATH environment variable. The installation command adds podman-compose to `/usr/bin`, which is already included in the `secure_path` option on the host.

c. If using Red Hat Enterprise Linux 8, verify that your Podman version is using NetAvark with Aardvark DNS instead of CNI.

i. Check to see if your networkBackend is set to CNI by running the following command:


```
podman info | grep networkBackend
```

- ii. If the `networkBackend` is set to `CNI`, you'll need to change it to `netavark`.
- iii. Install `netavark` and `aardvark-dns` using the following command:

```
dnf install aardvark-dns netavark
```

- iv. Open the `/etc/containers/containers.conf` file and modify the `network_backend` option to use "netavark" instead of "cni".

If `/etc/containers/containers.conf` doesn't exist, make the configuration changes to `/usr/share/containers/containers.conf`.

- v. Restart podman.

```
systemctl restart podman
```

- vi. Confirm `networkBackend` is now changed to "netavark" using the following command:

```
podman info | grep networkBackend
```

Docker Engine

Follow the documentation from Docker to install Docker Engine.

Steps

1. [View installation instructions from Docker](#)

Follow the steps to install a supported Docker Engine version. Do not install the latest version, as it is unsupported by the Console.

2. Verify that Docker is enabled and running.

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

Install the Console agent manually

Download and install the Console agent software on an existing Linux host on-premises.

Before you begin

You should have the following:

- Root privileges to install the Console agent.

- Details about a proxy server, if a proxy is required for internet access from the Console agent.

You have the option to configure a proxy server after installation but doing so requires restarting the Console agent.

- A CA-signed certificate, if the proxy server uses HTTPS or if the proxy is an intercepting proxy.



You cannot set a certificate for a transparent proxy server when manually installing the Console agent. If you need to set a certificate for a transparent proxy server, you must use the Maintenance Console after installation. Learn more about the [Agent Maintenance Console](#).

About this task

After installation, the Console agent automatically updates itself if a new version is available.

Steps

1. If the `http_proxy` or `https_proxy` system variables are set on the host, remove them:

```
unset http_proxy
unset https_proxy
```

If you don't remove these system variables, the installation fails.

2. Download the Console agent software and then copy it to the Linux host. You can download it either from the NetApp Console or the NetApp Support site.

- NetApp Console: Go to **Agents > Management > Deploy agent > On-prem > Manual install**.

Choose download the agent installer files or a URL to the files.

- NetApp Support Site (needed if you don't already have access to the Console) [NetApp Support Site](#),

3. Assign permissions to run the script.

```
chmod +x NetApp_Console_Agent_Cloud_<version>
```

Where <version> is the version of the Console agent that you downloaded.

4. If installing in a Government Cloud environment, disable the configuration checks. [Learn how to disable configuration checks for manual installations](#).
5. Run the installation script.

```
./NetApp_Console_Agent_Cloud_<version> --proxy <HTTP or HTTPS proxy
server> --cacert <path and file name of a CA-signed certificate>
```

You'll need to add proxy information if your network requires a proxy for internet access. You can add an explicit proxy during installation. The `--proxy` and `--cacert` parameters are optional and you won't be prompted to add them. If you have an explicit proxy server, you will need to enter the parameters as shown.



If you want to configure a transparent proxy, you can do so after you've installed. [Learn about the agent maintenance console](#)

+

Here is an example configuring an explicit proxy server with a CA-signed certificate:

+

```
./NetApp_Console_Agent_Cloud_v4.0.0--proxy
https://user:password@10.0.0.30:8080/ --cacert /tmp/cacert/certificate.cer
```

+

--proxy configures the Console agent to use an HTTP or HTTPS proxy server using one of the following formats:

+

```
* http://address:port
* http://user-name:password@address:port
* http://domain-name%92user-name:password@address:port
* https://address:port
* https://user-name:password@address:port
* https://domain-name%92user-name:password@address:port
```

+

Note the following:

+

The user can be a local user or domain user.

For a domain user, you must use the ASCII code for a \ as shown above.

The Console agent doesn't support user names or passwords that include the @ character.

If the password includes any of the following special characters, you must escape that special character by prepending it with a backslash: & or !

+

For example:

+

```
http://bxpproxyuser:netapp1\!@address:3128
```

1. If you used Podman, you'll need to adjust the aardvark-dns port.
 - a. SSH to the Console agent virtual machine.
 - b. Open podman /usr/share/containers/containers.conf file and modify the chosen port for Aardvark DNS service. For example, change it to 54.

```
vi /usr/share/containers/containers.conf
```

For example:

```
# Port to use for dns forwarding daemon with netavark in rootful
bridge
# mode and dns enabled.
# Using an alternate port might be useful if other DNS services
should
# run on the machine.
#
dns_bind_port = 54
```

- c. Reboot the Console agent virtual machine.

What's next?

You'll need to register the Console agent within the NetApp Console.

Register the Console agent with NetApp Console

Log into the Console and associate the Console agent with your organization. How you log in depends on the mode in which you are using Console. If you are using the Console in standard mode, you log in through the SaaS website. If you are using the Console in restricted mode, you log in locally from the Console agent host.

Steps

1. Open a web browser and enter the Console agent host URL:

The Console host URL can be a localhost, a private IP address, or a public IP address, depending on the configuration of the host. For example, if the Console agent 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 Console agent host.

2. Sign up or log in.
3. After you log in, set up the Console:
 - a. Specify the Console organization to associate with the Console agent.
 - b. Enter a name for the system.
 - c. Under **Are you running in a secured environment?** keep restricted mode disabled.

Restricted mode isn't supported when the Console agent is installed on-premises.

- d. Select **Let's start**.

Provide cloud provider credentials to NetApp Console

After you install and set up the Console agent, add your cloud credentials so that the Console agent has the required permissions to perform actions in AWS or Azure.

AWS

Before you begin

If you just created these AWS credentials, they may take a few minutes to become available. Wait a few minutes before you add the credentials to the Console.

Steps

1. Select **Administration > Credentials**.
2. Select **Organization credentials**.
3. Select **Add Credentials** and follow the steps in the wizard.
 - a. **Credentials Location**: Select *Amazon Web Services > Agent.
 - b. **Define Credentials**: Enter an AWS access key and secret key.
 - c. **Marketplace Subscription**: Associate a Marketplace subscription with these credentials by subscribing now or by selecting an existing subscription.
 - d. **Review**: Confirm the details about the new credentials and select **Add**.

You can now go to the [NetApp Console](#) to start using the Console agent.

Azure

Before you begin

If you just created these Azure credentials, they may take a few minutes to become available. Wait a few minutes before you add the credentials the Console agent.

Steps

1. Select **Administration > Credentials**.
2. Select **Add Credentials** and follow the steps in the wizard.
 - a. **Credentials Location**: Select **Microsoft Azure > Agent**.
 - b. **Define Credentials**: Enter information about the Microsoft Entra service principal that grants the required permissions:
 - Application (client) ID
 - Directory (tenant) ID
 - Client Secret
 - c. **Marketplace Subscription**: Associate a Marketplace subscription with these credentials by subscribing now or by selecting an existing subscription.
 - d. **Review**: Confirm the details about the new credentials and select **Add**.

Result

The Console agent now has the permissions that it needs to perform actions in Azure on your behalf. You can now go to the [NetApp Console](#) to start using the Console agent.

Install a Console agent on-premises using VCenter

If you are a VMWare user, you can use an OVA to install a Console agent in your VCenter. The OVA download or URL is available through the NetApp Console.



When you install a Console agent with your VCenter tools, you can use the VM web console to perform maintenance tasks. [Learn more about the VM console for the agent.](#)

Prepare to install the Console agent

Before installation, make sure your VM host meets the requirements and the Console agent can access the internet and targeted networks. To use NetApp data services or Cloud Volumes ONTAP, create cloud provider credentials for the Console agent to perform actions on your behalf.

Review Console agent host requirements

Make sure your host machine meets installation requirements before installing the Console agent.

- CPU: 8 cores or 8 vCPUs
- RAM: 32 GB
- Disk space: 165 GB (thick provisioned)
- vSphere 7.0 or higher
- ESXi host 7.03 or higher



Install the agent in a vCenter environment rather than directly on an ESXi host.

Set up network access for the Console agent

Work with your network administrator to ensure the Console agent has outbound access to the required endpoints and connections to targeted networks.

Connections to target networks

The Console agent requires a network connection to the location where you're planning to create and manage systems. For example, the network where you plan to create Cloud Volumes ONTAP systems or a storage system in your on-premises environment.

Outbound internet access

The network location where you deploy the Console agent must have an outbound internet connection to contact specific endpoints.

Endpoints contacted from computers when using the web-based NetApp Console

Computers that access the Console from a web browser must have the ability to contact several endpoints. You'll need to use the Console to set up the Console agent and for day-to-day use of the Console.

[Prepare networking for the NetApp console.](#)

Endpoints contacted from the Console agent

The Console agent requires outbound internet access to contact the following endpoints to manage resources and processes within your public cloud environment for day-to-day operations.

The endpoints listed below are all CNAME entries.



You can't manage resources in Google Cloud with an Console agent installed on your premises. To manage Google Cloud resources, install an agent in Google Cloud.

AWS

When the Console agent is installed on-premises, it needs network access to the following AWS endpoints in order to manage NetApp systems (such as Cloud Volumes ONTAP) deployed in AWS.

Endpoints contacted from the Console agent

The Console agent requires outbound internet access to contact the following endpoints to manage resources and processes within your public cloud environment for day-to-day operations.

The endpoints listed below are all CNAME entries.

Endpoints	Purpose
AWS services (amazonaws.com): <ul style="list-style-type: none">• CloudFormation• Elastic Compute Cloud (EC2)• Identity and Access Management (IAM)• Key Management Service (KMS)• Security Token Service (STS)• Simple Storage Service (S3)	To manage AWS resources. The endpoint depends on your AWS region. Refer to AWS documentation for details
Amazon FsX for NetApp ONTAP: <ul style="list-style-type: none">• api.workloads.netapp.com	The web-based console contacts this endpoint to interact with the Workload Factory APIs to manage and operate FSx for ONTAP based workloads.
https://mysupport.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support.
https://signin.b2c.netapp.com	To update NetApp Support Site (NSS) credentials or to add new NSS credentials to the NetApp Console.
https://support.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support as well as to receive software updates for Cloud Volumes ONTAP.
https://api.bluexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.bluexp.netapp.com https://cdn.auth0.com	To provide features and services within the NetApp Console.

Endpoints	Purpose
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	<p>To obtain images for Console agent upgrades.</p> <ul style="list-style-type: none"> When you deploy a new agent, the validation check tests connectivity to current endpoints. If you use previous endpoints, the validation check fails. To avoid this failure, skip the validation check. <p>Although the previous endpoints are still supported, NetApp recommends updating your firewall rules to the current endpoints as soon as possible. Learn how to update your endpoint list.</p> <ul style="list-style-type: none"> When you update to the current endpoints in your firewall, your existing agents will continue to work.

Azure

When the Console agent is installed on-premises, it needs network access to the following Azure endpoints in order to manage NetApp systems (such as Cloud Volumes ONTAP) deployed in Azure.

Endpoints	Purpose
https://management.azure.com https://login.microsoftonline.com https://blob.core.windows.net https://core.windows.net	To manage resources in Azure public regions.
https://management.chinacloudapi.cn https://login.chinacloudapi.cn https://blob.core.chinacloudapi.cn https://core.chinacloudapi.cn	To manage resources in Azure China regions.
https://mysupport.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support.
https://signin.b2c.netapp.com	To update NetApp Support Site (NSS) credentials or to add new NSS credentials to the NetApp Console.
https://support.netapp.com	To obtain licensing information and to send AutoSupport messages to NetApp support as well as to receive software updates for Cloud Volumes ONTAP.

Endpoints	Purpose
https://api.bluexp.netapp.com https://netapp-cloud-account.auth0.com https://netapp-cloud-account.us.auth0.com https://console.netapp.com https://components.console.bluexp.netapp.com https://cdn.auth0.com	To provide features and services within the NetApp Console.
https://bluexpinfraprod.eastus2.data.azurecr.io https://bluexpinfraprod.azurecr.io	To obtain images for Console agent upgrades. <ul style="list-style-type: none"> When you deploy a new agent, the validation check tests connectivity to current endpoints. If you use previous endpoints, the validation check fails. To avoid this failure, skip the validation check. <p>Although the previous endpoints are still supported, NetApp recommends updating your firewall rules to the current endpoints as soon as possible. Learn how to update your endpoint list.</p> <ul style="list-style-type: none"> When you update to the current endpoints in your firewall, your existing agents will continue to work.

Proxy server

NetApp supports both explicit and transparent proxy configurations. If you are using a transparent proxy, you only need to provide the certificate for the proxy server. If you are using an explicit proxy, you'll also need the IP address and credentials.

- IP address
- Credentials
- HTTPS certificate

Ports

There's no incoming traffic to the Console agent, unless you initiate it or if it is used as a proxy to send AutoSupport messages from Cloud Volumes ONTAP to NetApp Support.

- HTTP (80) and HTTPS (443) provide access to the local UI, which you'll use in rare circumstances.
- SSH (22) is only needed if you need to connect to the host for troubleshooting.
- Inbound connections over port 3128 are required if you deploy Cloud Volumes ONTAP systems in a subnet where an outbound internet connection isn't available.

If Cloud Volumes ONTAP systems don't have an outbound internet connection to send AutoSupport messages, the Console automatically configures those systems to use a proxy server that's included with the Console agent. The only requirement is to ensure that the Console agent's security group allows inbound connections over port 3128. You'll need to open this port after you deploy the Console agent.

Enable NTP

If you're planning to use NetApp Data Classification to scan your corporate data sources, you should enable a Network Time Protocol (NTP) service on both the Console agent and the NetApp Data Classification system so that the time is synchronized between the systems. [Learn more about NetApp Data classification](#)

Create Console agent cloud permissions for AWS or Azure

If you want to use NetApp data services in AWS or Azure with an on-premises Console agent, then you need to set up permissions in your cloud provider so that you can add the credentials to the Console agent after you install it.



You can't manage resources in Google Cloud with a Console agent installed on your premises. If you want to manage Google Cloud resources, you need to install an agent in Google Cloud.

AWS

For on-premises Console agents, provide AWS permissions by adding IAM user access keys.

Use IAM user access keys for on-premises Console agents; IAM roles are not supported for on-premises Console agents.

Steps

1. Log in to the AWS console and navigate to the IAM service.
2. Create a policy:
 - a. Select **Policies > Create policy**.
 - b. Select **JSON** and copy and paste the contents of the [IAM policy for the Console agent](#).
 - c. Finish the remaining steps to create the policy.

Depending on the NetApp data services that you're planning to use, you might need to create a second policy.

For standard regions, the permissions are spread across two policies. Two policies are required due to a maximum character size limit for managed policies in AWS. [Learn more about IAM policies for the Console agent](#).

3. Attach the policies to an IAM user.
 - [AWS Documentation: Creating IAM Roles](#)
 - [AWS Documentation: Adding and Removing IAM Policies](#)
4. Ensure that the user has an access key that you can add to the NetApp Console after you install the Console agent.

Result

You should now have IAM user access keys with the required permissions. After you install the Console agent, associate these credentials with the Console agent from the Console.

Azure

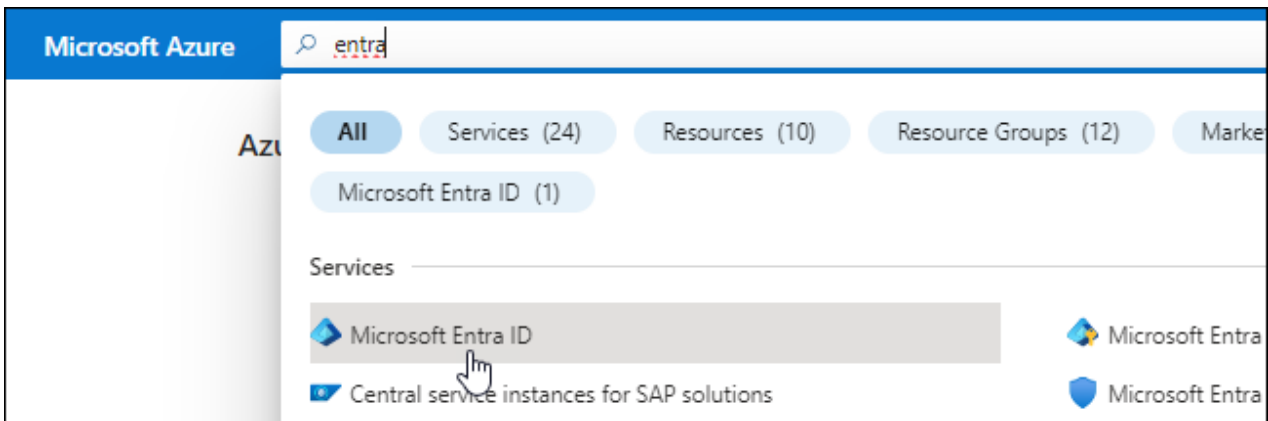
When the Console agent is installed on-premises, you need to give the Console agent Azure permissions by setting up a service principal in Microsoft Entra ID and getting the Azure credentials that the Console agent needs.

Create a Microsoft Entra application for role-based access control

1. Ensure that you have 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](#)

2. From the Azure portal, open the **Microsoft Entra ID** service.



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

You've created the AD application and service principal.

Assign the application to a role

1. Create a custom role:

Note that you can create an Azure custom role using the Azure portal, Azure PowerShell, Azure CLI, or REST API. The following steps show how to create the role using the Azure CLI. If you would prefer to use a different method, refer to [Azure documentation](#)

- a. Copy the contents of the [custom role permissions for the Console agent](#) and save them in a JSON file.
- 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.

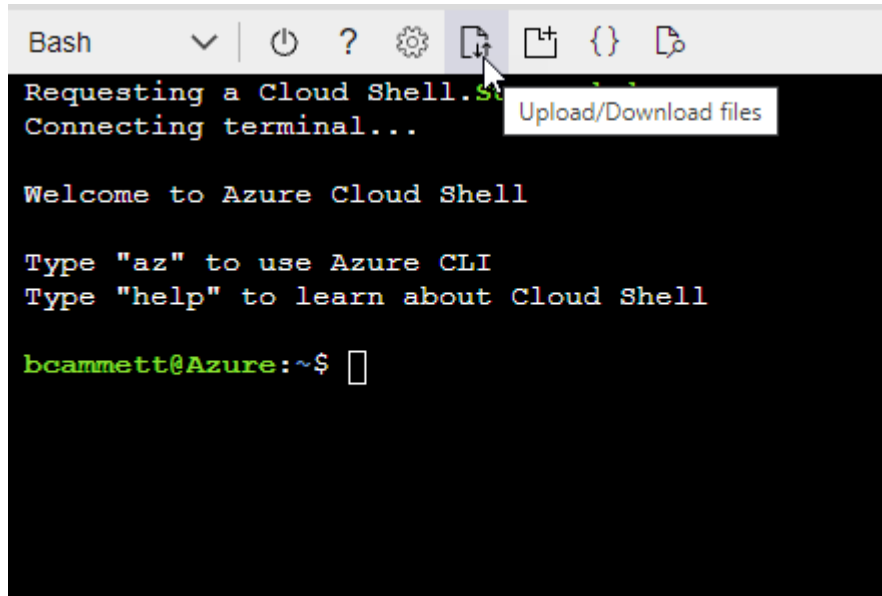
Example

```
"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 steps describe how to create the role by using Bash in Azure Cloud Shell.

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



- Use the Azure CLI to create the custom role:

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

You should now have a custom role called Console Operator that you can assign to the Console agent virtual machine.

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

Add role assignment ...

Got feedback?

Role **Members** Review + assign

Selected role Cloud Manager Operator 3.9.12_B

Assign access to ☒ User, group, or service principal
☐ Managed identity

Members + [Select members](#)

- Search for the name of the application.

Here's an example:

Select members ✕

Select ⓘ

test-service-principal

test-service-principal

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

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

If you want to deploy Cloud Volumes ONTAP from multiple Azure subscriptions, then you must bind the service principal to each of those subscriptions. In the NetApp Console, you can select the subscription that you want to use when deploying Cloud Volumes ONTAP.

Add Windows Azure Service Management API permissions

1. In the **Microsoft Entra ID** service, select **App registrations** and select the application.
2. Select **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. Select **Access Azure Service Management as organization users** and then select **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

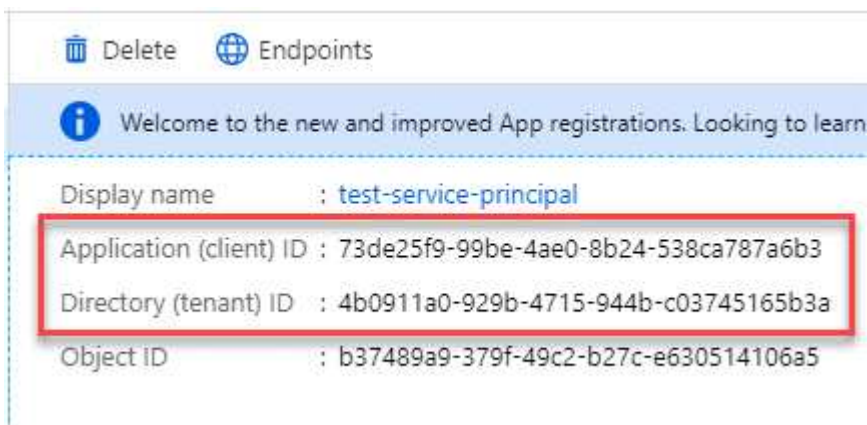


user_impersonation

Access Azure Service Management as organization users (preview) ⓘ

Get the application ID and directory ID for the application

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



When you add the Azure account to the Console, you need to provide the application (client) ID and the directory (tenant) ID for the application. The Console uses the IDs to programmatically sign in.

Create a client secret

1. Open the **Microsoft Entra ID** service.
2. Select **App registrations** and select your application.
3. Select **Certificates & secrets > New client secret**.
4. Provide a description of the secret and a duration.
5. Select **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

Install a Console agent in your VCenter environment

NetApp supports installing the Console agent in your VCenter environment. The OVA file includes a pre-configured VM image that you can deploy in your VMware environment. A file download or URL deployment is available directly from the NetApp Console. It includes the Console agent software and a self-signed certificate.

Download the OVA or copy the URL

Download the OVA or copy the OVA URL directly from the NetApp Console.

- 1. Select **Administration > Agents**.
- 2. On the **Overview** page, select **Deploy agent > On-Premises**.
- 3. Select **With OVA**.
- 4. Choose to either download the OVA or copy the URL to use in VCenter.

Deploy the agent in your VCenter

Log into your VCenter environment to deploy the agent.

Steps

- 1. Upload the self-signed certificate to your trusted certificates if your environment requires it. You replace this certificate after installation.[Learn how to replace the self-signed certificate.](#)
- 2. Deploy the OVA from the content library or local system.

From the local system	From the content library
a. Right-click and select Deploy OVF template....	a. Go to your content library and select the Console agent OVA.
b. Choose the OVA file from the URL or browse to its location, then select Next .	b. Select Actions > New VM from this template

- 3. Complete the Deploy OVF Template wizard to deploy the Console agent.
- 4. Select a name and folder for the VM, then select **Next**.
- 5. Select a compute resource, then select **Next**.
- 6. Review the details of the template, then select **Next**.
- 7. Accept the license agreement, then select **Next**.
- 8. Choose the type of proxy configuration you want to use: explicit proxy, transparent proxy, or no proxy.

9. Select the datastore where you want to deploy the VM, then select **Next**. Be sure it meets host requirements.
10. Select the network to which you want to connect the VM, then select **Next**. Ensure the network is IPv4 and has outbound internet access to the required endpoints.
11. In the **Customize template** window, complete the following fields:
 - **Proxy information**
 - If you selected explicit proxy, enter the proxy server hostname or IP address and port number, as well as the username, password.
 - If you selected transparent proxy, upload the respective certificate.
 - **Virtual Machine Configuration**
 - **Skip config check:** This check box is unchecked by default which means the agent runs a configuration check to validate network access.
 - NetApp recommends leaving this box unchecked so that the installation includes a configuration check of the agent. The Configuration check validates that the agent has network access to the required endpoints. If deployment fails because of connectivity issues, you can access the validation report and logs from the agent host. In some cases, if you are confident that the agent has network access, you can choose to skip the check. For example, if you are still using the [previous endpoints](#) used for agent upgrades, the validation fails with an error. To avoid this, mark the check box to install without a validation check. [Learn how to update your endpoint list](#).
 - **Maintenance password:** Set the password for the `maint` user that allows access to the agent maintenance console.
 - **NTP servers:** Specify one or more NTP servers for time synchronization.
 - **Hostname:** Set the hostname for this VM. It must not include the search domain. For example, an FQDN of `console10.searchdomain.company.com` should be entered as `console10`.
 - **Primary DNS:** Specify the primary DNS server to use for name resolution.
 - **Secondary DNS:** Specify the secondary DNS server to use for name resolution.
 - **Search domains:** Specify the search domain name to use when resolving the hostname. For example, if the FQDN is `console10.searchdomain.company.com`, then enter `searchdomain.company.com`.
 - **IPv4 address:** The IP address that is mapped to the hostname.
 - **IPv4 subnet mask:** The subnet mask for the IPv4 address.
 - **IPv4 gateway address:** The gateway address for the IPv4 address.
12. Select **Next**.
13. Review the details in the **Ready to complete** window, select **Finish**.

The vSphere task bar shows the progress as the Console agent is deployed.

14. Power on the VM.



If the deployment fails, you can access the validation report and logs from the agent host. [Learn how to troubleshoot installation issues](#).

Register the Console agent with NetApp Console

Log into the Console and associate the Console agent with your organization. How you log in depends on the mode in which you are using Console. If you are using the Console in standard mode, you log in through the SaaS website. If you are using the Console in restricted or private mode, you log in locally from the Console agent host.

Steps

1. Open a web browser and enter the Console agent host URL:

The Console host URL can be a localhost, a private IP address, or a public IP address, depending on the configuration of the host. For example, if the Console agent 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 Console agent host.

2. Sign up or log in.
3. After you log in, set up the Console:
 - a. Specify the Console organization to associate with the Console agent.
 - b. Enter a name for the system.
 - c. Under **Are you running in a secured environment?** keep restricted mode disabled.

Restricted mode isn't supported when the Console agent is installed on-premises.

- d. Select **Let's start**.

Add cloud provider credentials to the Console

After you install and set up the Console agent, add your cloud credentials so that the Console agent has the required permissions to perform actions in AWS or Azure.

AWS

Before you begin

If you just created these AWS credentials, they may take a few minutes to become available. Wait a few minutes before you add the credentials to the Console.

Steps

1. Select **Administration > Credentials**.
2. Select **Organization credentials**.
3. Select **Add Credentials** and follow the steps in the wizard.
 - a. **Credentials Location**: Select *Amazon Web Services > Agent.
 - b. **Define Credentials**: Enter an AWS access key and secret key.
 - c. **Marketplace Subscription**: Associate a Marketplace subscription with these credentials by subscribing now or by selecting an existing subscription.
 - d. **Review**: Confirm the details about the new credentials and select **Add**.

You can now go to the [NetApp Console](#) to start using the Console agent.

Azure

Before you begin

If you just created these Azure credentials, they may take a few minutes to become available. Wait a few minutes before you add the credentials the Console agent.

Steps

1. Select **Administration > Credentials**.
2. Select **Add Credentials** and follow the steps in the wizard.
 - a. **Credentials Location**: Select **Microsoft Azure > Agent**.
 - b. **Define Credentials**: Enter information about the Microsoft Entra service principal that grants the required permissions:
 - Application (client) ID
 - Directory (tenant) ID
 - Client Secret
 - c. **Marketplace Subscription**: Associate a Marketplace subscription with these credentials by subscribing now or by selecting an existing subscription.
 - d. **Review**: Confirm the details about the new credentials and select **Add**.

Result

The Console agent now has the permissions that it needs to perform actions in Azure on your behalf. You can now go to the [NetApp Console](#) to start using the Console agent.

Ports for the on-premises Console agent

The Console agent uses *inbound* ports when installed manually on an on-premises Linux host. Refer to these ports for planning purposes.

These inbound rules apply to all NetApp Console deployment modes.

Protocol	Port	Purpose
HTTP	80	<ul style="list-style-type: none">• Provides HTTP access from client web browsers to the local user interface• Used during the Cloud Volumes ONTAP upgrade process
HTTPS	443	Provides HTTPS access from client web browsers to the local user interface

Maintain Console agents

Maintain a VCenter or ESXi host for the Console agent

You can make changes to your existing VCenter or ESXi host after you deploy the Console agent. For example, you can increase the CPU or RAM of the VM instance that hosts the Console agent.

Perform these maintenance tasks using the VM web console:

- Increase disk size
- Restart the agent
- Update static routes
- Update search domains

Limitations

Upgrading the agent through the console is not yet supported. In addition, you can only view information about the IP address, DNS, and gateways.

Access the VM maintenance console

You can access the maintenance Console from the VSphere client.

Steps

1. Open the VSphere client and log in to your VCenter.
2. Select the VM instance that hosts the Console agent.
3. Select **Launch Web Console**.
4. Log in to the VM instance using the user name and password that you specified when you created the VM instance. The username is `maint` and the password is the one that you specified when you created the VM instance.

Change the maint user password

You can change the password for the `maint` user.

Steps

1. Open the VSphere client and log in to your VCenter.
2. Select the VM instance that hosts the Console agent.
3. Select **Launch Web Console**.

4. Log in to the VM instance using the user name and password that you specified when you created the VM instance. The username is `maint` and the password is the one that you specified when you created the VM instance.
5. Enter `1` to view the `System Configuration` menu.
6. Enter `1` to change the maintenance user password and follow the on-screen prompts.

Increase the CPU or RAM of the VM instance

You can increase the CPU or RAM of the VM instance that hosts the Console agent.

Edit the VM instance settings in your VCenter or ESXi host, then use the maintenance Console to apply the changes.

Steps in the VSphere client

1. Open the VSphere client and log in to your VCenter.
2. Select the VM instance that hosts the Console agent.
3. Right-click the VM instance and select **Edit Settings**.
4. Increase the hard drive space used for `/opt` or the `/var` partition.
 - a. Select **Hard Disk 2** to increase the hard drive space used for `/opt`.
 - b. Select **Hard Disk 3** to increase the hard drive space used for `/var`.
5. Save your changes.

Steps in the maintenance console

1. Open the VSphere client and log in to your VCenter.
2. Select the VM instance that hosts the Console agent.
3. Select **Launch Web Console**.
4. Log in to the VM instance using the user name and password that you specified when you created the VM instance. The username is `maint` and the password is the one that you specified when you created the VM instance.
5. Enter `1` to view the `System Configuration` menu.
6. Enter `2` and follow the on-screen prompts. The console scans for new settings and increases the size of the partitions.

View network settings for the agent VM

View the network settings for the agent VM in the VSphere client to confirm or troubleshoot network issues. You can only view (not update) the following network settings: IP address and DNS details.

Steps

1. Open the VSphere client and log in to your VCenter.
2. Select the VM instance that hosts the Console agent.
3. Select **Launch Web Console**.
4. Log in to the VM instance using the user name and password that you specified when you created the VM instance. The username is `maint` and the password is the one that you specified when you created the VM instance.

5. Enter 2 to view the `Network Configuration` menu.
6. Enter a number between 1 and 6 to view the corresponding network settings.

Update the static routes for the agent VM

Add, update, or remove static routes for the agent VM as needed.

Steps

1. Open the VSphere client and log in to your VCenter.
2. Select the VM instance that hosts the Console agent.
3. Select **Launch Web Console**.
4. Log in to the VM instance using the user name and password that you specified when you created the VM instance. The username is `maint` and the password is the one that you specified when you created the VM instance.
5. Enter 2 to view the `Network Configuration` menu.
6. Enter 7 to update static routes and follow the on-screen prompts.
7. Press Enter.
8. Optionally, make additional changes.
9. Enter 9 to commit your changes.

Update domain search settings for the agent VM

You can update the search domain settings for the agent VM.

Steps

1. Open the VSphere client and log in to your VCenter.
2. Select the VM instance that hosts the Console agent.
3. Select **Launch Web Console**.
4. Log in to the VM instance using the user name and password that you specified when you created the VM instance. The username is `maint` and the password is the one that you specified when you created the VM instance.
5. Enter 2 to view the `Network Configuration` menu.
6. Enter 8 to update the domain search settings and follow the on-screen prompts.
7. Press Enter.
8. Optionally, make additional changes.
9. Enter 9 to commit your changes.

Access the agent diagnostic tools

Access diagnostic tools to troubleshoot issues with the Console agent. NetApp Support may ask you to do this when troubleshooting issues.

Steps

1. Open the VSphere client and log in to your VCenter.

2. Select the VM instance that hosts the Console agent.
3. Select **Launch Web Console**.
4. Log in to the VM instance using the user name and password that you specified when you created the VM instance. The username is `maint` and the password is the one that you specified when you created the VM instance.
5. Enter 3 to view the Support and Diagnostics menu.
6. Enter 1 to access the diagnostic tools and follow the on-screen prompts.
 - + For example, you can verify that all agent services are running. [Check the Console agent status](#).

Access the agent diagnostic tools remotely

You can access diagnostic tools remotely with a tool such as Putty. Enable SSH access to the agent VM by assigning a one-time password.

SSH access enables advanced terminal features like copy and paste.

Steps

1. Open the VSphere client and log in to your VCenter.
2. Select the VM instance that hosts the Console agent.
3. Select **Launch Web Console**.
4. Log in to the VM instance using the user name and password that you specified when you created the VM instance. The username is `maint` and the password is the one that you specified when you created the VM instance.
5. Enter 3 to view the `Support and Diagnostics` menu.
6. Enter 2 to access the diagnostic tools and follow the on-screen prompts to configure a one-time password that expires in 24 hours.
7. Use an SSH tool such as Putty to connect to the agent VM using the user name `diag` and the one-time password that you configured.

Install a CA-signed certificate for web-based console access

When you use the NetApp Console in restricted mode, the user interface is accessible from the Console agent virtual machine that's deployed in your cloud region or on-premises. By default, the Console uses a self-signed SSL certificate to provide secure HTTPS access to the web-based console running on the Console agent.

If required by your business, you can install a certificate signed by a certificate authority (CA), which provides better security protection than a self-signed certificate. After you install the certificate, the Console uses the CA-signed certificate when users access the web-based console.

Install an HTTPS certificate

Install a certificate signed by a CA for secure access to the web-based console running on the Console agent.

About this task

You can install the certificate using one of the following options:

- Generate a certificate signing request (CSR) from the Console, submit the certificate request to a CA, and

then install the CA-signed certificate on the Console agent.

The key pair that the Console uses to generate the CSR is stored internally on the Console agent. The Console automatically retrieves the same key pair (private key) when you install the certificate on the Console agent.

- Install a CA-signed certificate that you already have.

With this option, the CSR is not generated through the Console. You generate the CSR separately and store the private key externally. You provide the Console with the private key when you install the certificate.

Steps

1. Select **Administration > Agents**.
2. On the **Overview** page, select the action menu for a Console agent and select **HTTPS Setup**.

The Console agent must be connected to edit it.

3. In the HTTPS Setup page, install a certificate by generating a certificate signing request (CSR) or by installing your own CA-signed certificate:

Option	Description
Generate a CSR	<ol style="list-style-type: none">a. Enter the host name or DNS of the Console agent host (its Common Name), and then select Generate CSR. The Console displays a certificate signing request.b. Use the CSR to submit an SSL certificate request to a CA. The certificate must use the Privacy Enhanced Mail (PEM) Base-64 encoded X.509 format.c. Upload the certificate file and then select Install.
Install your own CA-signed certificate	<ol style="list-style-type: none">a. Select Install CA-signed certificate.b. Load both the certificate file and the private key and then select Install. The certificate must use the Privacy Enhanced Mail (PEM) Base-64 encoded X.509 format.

Result

The Console agent now uses the CA-signed certificate to provide secure HTTPS access. The following image shows an agent that is configured for secure access:

HTTPS Certificate

Change Certificate

✓ HTTPS Setup is active

Expiration: Aug 15, 2029 10:09:01 am

Issuer: C=IL, ST=Israel, L=Tel Aviv, O=NetApp, OU=Dev, CN= Localhost, E=Admin@netapp.com

Subject: C=IL, ST=Israel, L=Tel Aviv, O=NetApp, OU=Dev, CN= Localhost, E=Admin@netapp.com

Certificate:

View CSR

Renew the Console HTTPS certificate

You should renew the agent's HTTPS certificate before it expires to ensure secure access. If you don't renew the certificate before it expires, a warning appears when users access the web console using HTTPS.

Steps

1. Select **Administration > Agents**.
2. On the **Overview** page, select the action menu for a Console agent and select **HTTPS Setup**.

Details about the certificate displays, including the expiration date.

3. Select **Change Certificate** and follow the steps to generate a CSR or install your own CA-signed certificate.

Configure a Console agent to use a proxy server

If your corporate policies require you to use a proxy server for all communication to the internet, then you need to configure your agents to use that proxy server. If you didn't configure a Console agent to use a proxy server during installation, then you can configure the Console agent to use that proxy server at any time.

The agent's proxy server enables outbound internet access without a public IP or NAT gateway. The proxy server provides outbound connectivity only for the Console agent, not for Cloud Volumes ONTAP systems.

If Cloud Volumes ONTAP systems lack outbound internet access, the Console configures them to use the Console agent's proxy server. You must ensure that the Console agent's security group allows inbound connections over port 3128. Open this port after deploying the Console agent.

If the Console agent itself doesn't have an outbound internet connection, Cloud Volumes ONTAP systems cannot use the configured proxy server.

Supported configurations

- Transparent proxy servers are supported for agents that serve Cloud Volumes ONTAP systems. If you use NetApp data services with Cloud Volumes ONTAP, create a dedicated agent for Cloud Volumes ONTAP where you can use a transparent proxy server.
- Explicit proxy servers are supported with all agents, including those that manage Cloud Volumes ONTAP systems and those that manage NetApp data services.
- HTTP and HTTPS.
- The proxy server can reside in the cloud or in your network.



Once you have configured a proxy, you cannot change the proxy type. If you need to change the proxy type, you remove the Console agent and add a new agent with the new proxy type.

Enable an explicit proxy on a Console agent

When you configure a Console agent to use a proxy server, that agent and the Cloud Volumes ONTAP systems that it manages (including any HA mediators), all use the proxy server.

This operation restarts the Console agent. Verify the Console agent is idle before proceeding.

Steps

1. Select **Administration > Agents**.
2. On the **Overview** page, select the action menu for a Console agent and select **Edit agent**.

The Console agent must be active to edit it.

3. Select **HTTP Proxy Configuration**.
4. Select **Explicit proxy** in the Configuration type field.
5. Select **Enable Proxy**.
6. Specify the server using the syntax `http://address:port` or `https://address:port`
7. Specify a user name and password if basic authentication is required for the server.

Note the following:

- The user can be a local user or domain user.
- For a domain user, you must enter the ASCII code for the \ as follows: `domain-name%92user-name`

For example: `netapp%92proxy`

- The Console doesn't support passwords that include the @ character.

8. Select **Save**.

Enable a transparent proxy for a Console agent

Only Cloud Volumes ONTAP supports using a transparent proxy on the Console agent. If you use NetApp data services in addition to Cloud Volumes ONTAP, you should create a separate agent to use for data services or to use for Cloud Volumes ONTAP.

Before enabling a transparent proxy, ensure that the following requirements are met:

- The agent is installed on the same network as the transparent proxy server.
- TLS inspection is enabled on the proxy server.
- You have a certificate in PEM format that matches the one used on the transparent proxy server.
- You do not use the Console agent for any NetApp data services other than Cloud Volumes ONTAP.

To configure an existing agent to use a transparent proxy server, you use the Console agent maintenance tool that is available through the command line on the Console agent host.

When you configure a proxy server, the Console agent restarts. Verify the Console agent is idle before proceeding.

Steps

Ensure that you have a certificate file in PEM format for the proxy server. If you do not have a certificate, contact your network administrator to obtain one.

1. Open a command-line interface on the Console agent host.
2. Navigate to the Console agent maintenance tool directory: `/opt/application/netapp/service-manager-2/agent-maint-console`
3. Run the following command to enable the transparent proxy, where `/home/ubuntu/<certificate-file>.pem` is the directory and name certificate file that you have for the proxy server:

```
./agent-maint-console proxy add -c /home/ubuntu/<certificate-file>.pem
```

Ensure that the certificate file is in PEM format and resides in the same directory as the command or specify the full path to the certificate file.

```
./agent-maint-console proxy add -c /home/ubuntu/<certificate-file>.pem
```

Modify the transparent proxy for the Console agent

You can update a Console agent's existing transparent proxy server by using the `proxy update` command or remove the transparent proxy server by using the `proxy remove` command. For more information, review the documentation for [Agent maintenance console](#).



Once you have configured a proxy, you cannot change the proxy type. If you need to change the proxy type, you remove the Console agent and add a new agent with the new proxy type.

Update the Console agent proxy if it loses access to the internet

If the proxy configuration for your network changes, your agent might lose access to the internet. For example, if someone changes the password for the proxy server or updates the certificate. In this case, you'll need to access the UI from the Console agent host directly and update the settings. Ensure you have network access to the Console agent host and that you can log into the Console.

Enable direct API traffic

If you configured a Console agent to use a proxy server, you can enable direct API traffic on the Console agent

in order to send API calls directly to cloud provider services without going through the proxy. Agents running in AWS, Azure, or Google Cloud support this option.

If you disable Azure Private Links with Cloud Volumes ONTAP and use service endpoints, enable direct API traffic. Otherwise, the traffic won't be routed properly.

[Learn more about using an Azure Private Link or service endpoints with Cloud Volumes ONTAP](#)

Steps

1. Select **Administration > Agents**.
2. On the **Overview** page, select the action menu for a Console agent and select **Edit agent**.

The Console agent must be active to edit it.

3. Select **Support Direct API Traffic**.
4. Select the checkbox to enable the option and then select **Save**.

Troubleshoot the Console agent

To troubleshoot issues with a Console agent, you can verify issues yourself or work with NetApp Support who might ask for your system ID, agent version, or the latest AutoSupport messages.

If you have a NetApp Support Site account, you can also view the [NetApp Knowledge Base](#).

Common error messages and resolutions

This table lists common error messages and shows how to fix them:

Error message	Explanation	What to do
Unable to load the Console agent UI	Agent installation has failed	<ul style="list-style-type: none">• Verify that the Service Manager service is active.• Verify that all containers are running.• Ensure your firewall allows access to the service at port 8888.• If you still have problems, contact support.
Cannot access the NetApp agent UI	This message appears when trying to access the IP address of an agent. The agent can fail to initialize if it doesn't have the correct network access or if it is unstable.	<ul style="list-style-type: none">• Connect to the Console agent.• Verify that the Service Manager service• Verify that the agent has the network access it needs. Learn more about required network access endpoints.

Error message	Explanation	What to do
Unable to load agent settings	The Console displays this message when you try to access the Agent settings page..	<ul style="list-style-type: none"> • Check if the OCCM container is running and working. • If the issue persists, contact support.
Unable to load support information for the agent.	This message displays if the agent cannot access your support account.	<ul style="list-style-type: none"> • Verify that the agent has outbound access to the required endpoints. Learn more about required network access endpoints.

Check the Console agent status

Use one of the following commands to verify your Console agent. All services should have a status of *Running*. If this isn't the case, contact NetApp support.



For more detailed information about accessing the Console agent diagnostics, see the following topics:

- [Check the Console agent status \(for Linux host deployments\)](#)
- [Check the Console agent status \(for VCenter deployments\)](#)

Docker (for Ubuntu and VCenter deployments)

```
docker ps -a
```

Podman (for RedHat Enterprise Linux deployments)

```
podman ps -a
```

View the Console agent version

View the Console agent version to confirm the upgrade or share it with your NetApp representative.

Steps

1. Select **Administration > Support > Agents**.

The Console displays the version at the top of the page.

Verify network access

Ensure that the Console agent has the network access it needs. [Learn more about required network access points.](#)

Run configuration checks on the Console agent

Run configuration checks on Console agents from the Console or the Agent maintenance console to make sure they are connected.

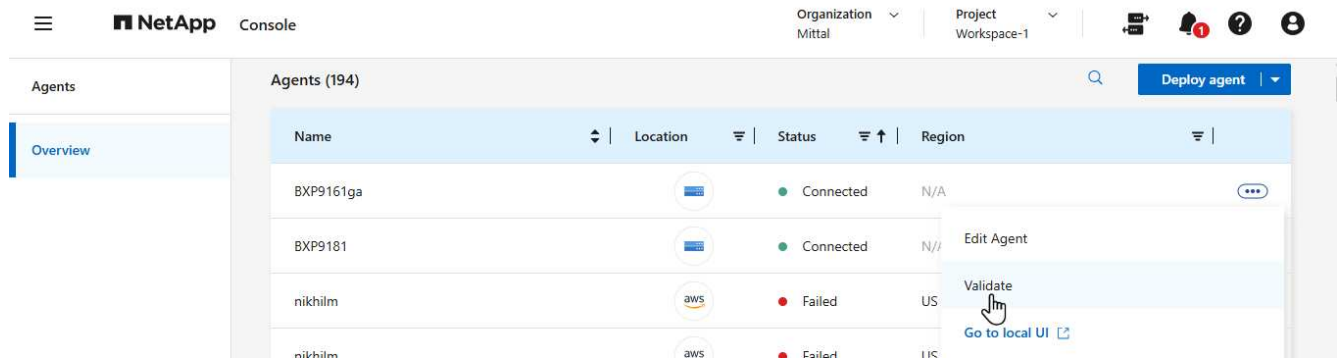
You can also run configuration checks using the agent maintenance console. [Learn more about using the config-checker validate command.](#)



You can only validate agents that have a status of **Connected**.

Steps from the Console

1. Select **Administration > Agents**.
2. Select the action menu for a Console agent that you want to check and choose **Validate**.



Validation can take up to 15 minutes. Results show when it is done.

Console agent installation issues

If the installation fails, view the report and logs to resolve the issues.

You can also access the validation report in JSON format and the configuration logs directly from the Console agent host in the following directories:

```
/tmp/netapp-console-agents/logs  
  
/tmp/netapp-console-agents/results.json
```



- For new agent deployments, NetApp checks for the following endpoints: [listed here](#). This configuration check fails with an error if you are using the previous endpoints used for upgrades, [listed here](#). NetApp recommends updating your firewall rules to allow access to the current endpoints and block access to the previous endpoints at your earliest convenience [Learn how to update your networking](#).
- If you update the endpoints in your firewall, your existing agents will continue to work.

Disable configuration checks for manual installations

There may be times when you need to disable the configuration checks that verify outbound connectivity during installation. For example, when manually installing an agent in your Government Cloud environment, you need to disable the configuration checks or the installation will fail.

Steps

You disable the configuration check by setting the `skipConfigCheck` flag in the `com/opt/application/netapp/service-manager-2/config.json` file. By default, this flag is set to false and the

configuration check verifies outbound access for the agent. Set this flag to true to disable the check. Be familiar with JSON syntax before completing this step.

To re-enable the configuration check, use these steps and set the *skipConfigCheck* flag to false.

Steps

1. Access the Console agent host as root or with sudo privileges.
2. Create a backup copy of the */opt/application/netapp/service-manager-2/config.json* file to ensure you can revert your changes.
3. Stop the service manager 2 service by running the following command:

```
systemctl stop netapp-service-manager.service
```

1. Edit the */opt/application/netapp/service-manager-2/config.json* file and change the value of the *skipConfigCheck* flag to true.

```
"skipConfigCheck": true
```

2. Save your file.
3. Restart the service manager 2 service by running the following command:

```
systemctl restart netapp-service-manager.service
```

Work with NetApp Support

If you haven't been able to resolve the issues with your Console agent, you may want to contact NetApp Support. NetApp support may ask for the Console agent ID or for you to send the Console agent logs to them if they don't have them already.

Find the Console agent ID







To help you get started, you may need the system ID of your Console agent. The ID is typically used for licensing and troubleshooting purposes.

Steps

1. Select **Administration > Support > Agents**.

You can find the system ID at the top of the page.

Example

 staging-onprem-connector Agent name	 3.9.56 / 875 Version/Build	 netapp Company
 4mcQIG1xzzDEhGq0CgorxV1Da...	 a39d460d-a64e-47e2-b066-ac...	 2da1c40131a6 Server Name
Client ID	System ID	

2. Hover and click on the ID to copy it.

Download or send an AutoSupport message

If you're having problems, NetApp might ask you to send an AutoSupport message to NetApp support for troubleshooting purposes.



The NetApp Console takes up to five hours to send AutoSupport messages due to load balancing. For urgent communication, download the file and send it manually.

Steps

1. Select **Administration > Support > Agents**.
2. Depending on how you need to send the information to NetApp support, choose one of the following options:
 - a. Select the option to download the AutoSupport message to your local machine. You can then send it to NetApp Support using a preferred method.
 - b. Select **Send AutoSupport** to directly send the message to NetApp Support.

Fix download failures when using a Google Cloud NAT gateway

The Console agent automatically downloads software updates for Cloud Volumes ONTAP. Your configuration can cause the download to fail if it uses a Google Cloud NAT gateway. You can correct this issue by limiting the number of parts that the software image is divided into. This step must be completed by using the API.

Step

1. Submit a PUT request to `/occm/config` with the following JSON as body:

```
{
  "maxDownloadSessions": 32
}
```

The value for `maxDownloadSessions` can be 1 or any integer greater than 1. If the value is 1, then the downloaded image will not be divided.

Note that 32 is an example value. The value depends on your NAT configuration and the number of simultaneous sessions.

[Learn more about the /occm/config API call](#)

[View troubleshooting information created by the NetApp Support team.](#)

Uninstall and remove a Console agent

Uninstall the a Console agent to troubleshoot issues or to permanently remove it from the host. The steps that you need to use depends on the deployment mode that you're using. Once you have removed a Console agent from your environment, you can remove it from the Console.

[Learn about NetApp Console deployment modes.](#)

Uninstall the agent when using standard or restricted mode

If you're using standard mode or restricted mode (in other words, the agent host has outbound connectivity), then you should follow the steps below to uninstall the agent.

Steps

1. Connect to the Linux VM for the agent.
2. From the Linux host, run the uninstallation script:

```
/opt/application/netapp/service-manager-2/uninstall.sh [silent]
```

silent runs the script without prompting you for confirmation.

Remove Console agents from the Console

If you have deleted an agent VM or uninstalled the agent, you should remove it from the list of agents in the Console. After you delete an agent VM or uninstall the agent software, the agent shows a status of **Disconnected** in the Console.

Note the following about removing a Console agent:

- This action doesn't delete the virtual machine.
- This action can't be reverted—once you remove a Console agent, you can't add it back.

Steps

1. Select **Administration > Agents**.
2. On the **Overview** page , select the action menu for a disconnected agent and select **Remove agent**.
3. Enter the name of the agent to confirm and then select **Remove**.

Manage cloud provider credentials

AWS

[Learn about AWS credentials and permissions in NetApp Console](#)

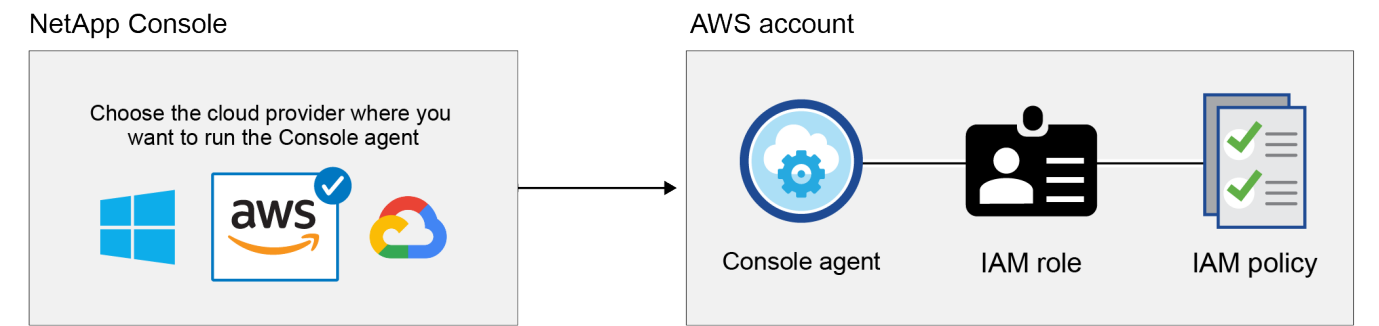
You manage AWS credentials and marketplace subscriptions directly from NetApp Console to ensure secure deployment of Cloud Volumes ONTAP and other data services by providing appropriate IAM credentials during Console agent deployment and

associating them with AWS Marketplace subscriptions for billing.

Initial AWS credentials

When you deploy an Console agent from the Console, you need to provide the ARN of an IAM role or access keys for an IAM user. The authentication method must have permissions to deploy the Console agent in AWS. The required permissions are listed in the [Agent deployment policy for AWS](#).

When the Console launches the Console agent in AWS, it creates an IAM role and a profile for the agent. It also attaches a policy that provides the Console agent with permissions to manage resources and processes within that AWS account. [Review how the Agent uses the permissions](#).



If you add a new Cloud Volumes ONTAP system, the Console selects these AWS credentials by default:

Details & Credentials			
Instance Profile		QA Subscription	Edit Credentials
Credentials	Account ID	Marketplace Subscription	

Deploy all of your Cloud Volumes ONTAP systems using the initial AWS credentials, or you can add additional credentials.

Additional AWS credentials

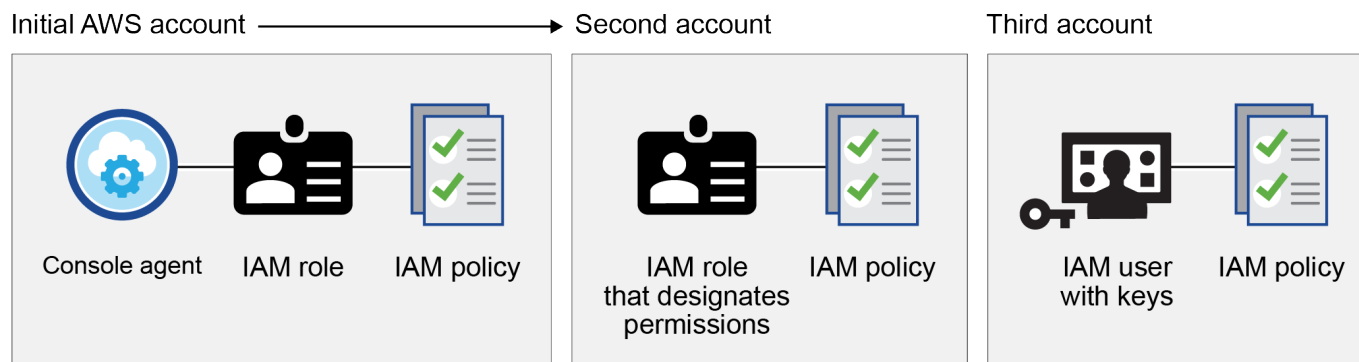
You might add additional AWS credentials to the Console in the following cases:

- To use your existing Console agent with an additional AWS account
- To create a new agent in a specific AWS account
- To create and manage FSx for ONTAP file systems

Review the sections below for more details.

Add AWS credentials to use a Console agent with another AWS account

To use the Console with additional AWS accounts, provide AWS keys or the ARN of a role in a trusted account. The following image shows two additional accounts, one providing permissions through an IAM role in a trusted account and another through the AWS keys of an IAM user:



You add account credentials to the Console by specifying the Amazon Resource Name (ARN) of IAM role or the AWS keys for the IAM user.

For example, you can switch between credentials when creating a new Cloud Volumes ONTAP system:

The screenshot shows the 'Edit Credentials & Add Subscription' dialog. It includes a section for 'Associate Subscription to Credentials' with a list of credentials. The list shows 'keys | Account ID: [redacted]', 'Instance Profile | Account ID: [redacted]', and 'casaba QA subscription'. There is an 'Add Subscription' button and 'Apply' and 'Cancel' buttons at the bottom.

[Learn how to add AWS credentials to an existing agent.](#)

Add AWS credentials to create a Console agent

Adding AWS credentials provides permissions to create a Console agent.

[Learn how to add AWS credentials to the Console for creating a Console agent](#)

Add AWS credentials for FSx for ONTAP

Add AWS credentials to the Console to provide the necessary permissions to create and manage an FSx for ONTAP system.

[Learn how to add AWS credentials to the Console for Amazon FSx for ONTAP](#)

Credentials and marketplace subscriptions

You must associate the credentials that you add to a Console agent with an AWS Marketplace subscription to pay for Cloud Volumes ONTAP at an hourly rate (PAYGO) and other NetApp data services or through an annual contract.

[Learn how to associate an AWS subscription.](#)

Note the following about AWS credentials and marketplace subscriptions:

- You can associate only one AWS Marketplace subscription with a set of AWS credentials
- You can replace an existing marketplace subscription with a new subscription

FAQ

The following questions are related to credentials and subscriptions.

How can I securely rotate my AWS credentials?

As described in the sections above, the Console enables you to provide AWS credentials in a few ways: an IAM role associated with the Console agent, by assuming an IAM role in a trusted account, or by providing AWS access keys.

With the first two options, the Console uses the AWS Security Token Service to obtain temporary credentials that rotate constantly. This process is the best practice—it's automatic and secure.

If you provide the Console with AWS access keys, you should rotate the keys by updating them in the Console at a regular interval. This is a completely manual process.

Can I change the AWS Marketplace subscription for Cloud Volumes ONTAP systems?

Yes, you can. When you change the AWS Marketplace subscription that's associated with a set of credentials, all existing and new Cloud Volumes ONTAP systems are charged against the new subscription.

[Learn how to associate an AWS subscription.](#)

Can I add multiple AWS credentials, each with different marketplace subscriptions?

All AWS credentials that belong to the same AWS account will be associated with the same AWS Marketplace subscription.

If you have multiple AWS credentials that belong to different AWS accounts, then those credentials can be associated with the same AWS Marketplace subscription or with different subscriptions.

Can I move existing Cloud Volumes ONTAP systems to a different AWS account?

No, it's not possible to move the AWS resources associated with your Cloud Volumes ONTAP system to a different AWS account.

How do credentials work for marketplace deployments and on-premises deployments?

The sections above describe the recommended deployment method for the Console agent, which is from the Console. You can also deploy an agent in AWS from the AWS Marketplace and you can manually install the Console agent software on your own Linux host or in your VCenter.

If you use the Marketplace, permissions are provided in the same way. You just need to manually create and

set up the IAM role, and then provide permissions for any additional accounts.

For on-premises deployments, you can't set up an IAM role for the Console, but you can provide permissions using AWS access keys.

To learn how to set up permissions, refer to the following pages:

- Standard mode
 - [Set up permissions for an AWS Marketplace deployment](#)
 - [Set up permissions for on-premises deployments](#)
- Restricted mode
 - [Set up permissions for restricted mode](#)

Manage AWS credentials and marketplace subscriptions for NetApp Console

Add and manage AWS credentials so that you deploy and manage cloud resources in your AWS accounts from the NetApp Console. If you manage multiple AWS Marketplace subscriptions, you can assign each one of them to different AWS credentials from the Credentials page.

Overview

You can add AWS credentials to an existing Console agent or directly to the Console:

- Add additional AWS credentials to an existing agent

Add AWS credentials to a Console agent to manage cloud resources. [Learn how to add AWS credentials to a Console agent.](#)

- Add AWS credentials to the Console for creating a Console agent

Adding new AWS credentials to the Console provides the permissions needed to create a Console agent. [Learn how to add AWS credentials to the NetApp Console.](#)

- Add AWS credentials to the Console for FSx for ONTAP

Add new AWS credentials to the Console to create and manage FSx for ONTAP. [Learn how to set up permissions for FSx for ONTAP](#)

How to rotate credentials

The NetApp Console enables you to provide AWS credentials in a few ways: an IAM role associated with the agent instance, by assuming an IAM role in a trusted account, or by providing AWS access keys. [Learn more about AWS credentials and permissions.](#)

With the first two options, the Console uses the AWS Security Token Service to obtain temporary credentials that rotate constantly. This process is the best practice because it's automatic and it's secure.

Manually rotate AWS access keys by updating them in the Console.

Add additional credentials to a Console agent

Add additional AWS credentials to a Console agent so that it has the permissions needed to manage resources and processes within your public cloud environment. You can either provide the ARN of an IAM role in another account or provide AWS access keys.

[Learn how the NetApp Console uses AWS credentials and permissions.](#)

Grant permissions

Grant permissions before adding AWS credentials to a Console agent. The permissions allow a Console agent to manage resources and processes within that AWS account. You can provide the permissions with the ARN of a role in a trusted account or AWS keys.



If you deployed a Console agent from the Console, it automatically added AWS credentials for the account in which you deployed a Console agent. This ensures the necessary permissions are in place for managing resources.

Choices

- [Grant permissions by assuming an IAM role in another account](#)
- [Grant permissions by providing AWS keys](#)

Grant permissions by assuming an IAM role in another account

You can set up a trust relationship between the source AWS account in which you deployed a Console agent and other AWS accounts by using IAM roles. You would then provide the Console with the ARN of the IAM roles from the trusted accounts.

If a Console agent is installed on-premises, you can't use this authentication method. You must use AWS keys.

Steps

1. Go to the IAM console in the target account in which you want to provide a Console agent with permissions.
2. Under Access Management, select **Roles > Create Role** and follow the steps to create the role.

Be sure to do the following:

- Under **Trusted entity type**, select **AWS account**.
 - Select **Another AWS account** and enter the ID of the account where a Console agent instance resides.
 - Create the required policies by copying and pasting the contents of [the IAM policies for a Console agent](#).
3. Copy the Role ARN of the IAM role so that you can paste it in the Console later on.

Result

The account has the required permissions. [You can now add the credentials to a Console agent.](#)

Grant permissions by providing AWS keys

If you want to provide the Console with AWS keys for an IAM user, then you need to grant the required permissions to that user. The the Console IAM policy defines the AWS actions and resources that the Console

is allowed to use.

You must use this authentication method if a Console agent is installed on-premises. You can't use an IAM role.

Steps

1. From the IAM console, create policies by copying and pasting the contents of [the IAM policies for a Console agent](#).

[AWS Documentation: Creating IAM Policies](#)

2. Attach the policies to an IAM role or an IAM user.
 - [AWS Documentation: Creating IAM Roles](#)
 - [AWS Documentation: Adding and Removing IAM Policies](#)

Add the credentials to an existing agent

After you provide an AWS account with the required permissions, you can add the credentials for that account to an existing agent. This enables you to launch Cloud Volumes ONTAP systems in that account using the same agent.



New credentials in your cloud provider may take a few minutes to become available.

Steps

1. Use the top navigation bar to select a Console agent to which you want to add credentials.
2. In the left navigation bar, select **Administration > Credentials**.
3. On the **Organization credentials** page, select **Add Credentials** and follow the steps in the wizard.
 - a. **Credentials Location:** Select **Amazon Web Services > Agent**.
 - b. **Define Credentials:** Provide the ARN (Amazon Resource Name) of a trusted IAM role, or enter an AWS access key and secret key.
 - c. **Marketplace Subscription:** Associate a Marketplace subscription with these credentials by subscribing now or by selecting an existing subscription.

To pay for services at an hourly rate (PAYGO) or with an annual contract, you must associate AWS credentials with your AWS Marketplace subscription.

- d. **Review:** Confirm the details about the new credentials and select **Add**.

Result

You can now switch to a different set of credentials from the Details and Credentials page when adding a subscription to the Console.

Edit Credentials & Add Subscription

Associate Subscription to Credentials ?

Credentials

keys | Account ID:

Instance Profile | Account ID:

casaba QA subscription

+ Add Subscription

Apply

Cancel

Add credentials to the Console for creating a Console agent

Add AWS credentials by providing the ARN of an IAM role that gives the permissions needed to create a Console agent. You can choose these credentials when creating a new agent.

Set up the IAM role

Set up an IAM role that enables the NetApp Console software as a service (SaaS) layer to assume the role.

Steps

1. Go to the IAM console in the target account.
2. Under Access Management, select **Roles > Create Role** and follow the steps to create the role.

Be sure to do the following:

- Under **Trusted entity type**, select **AWS account**.
- Select **Another AWS account** and enter the ID of the NetApp Console SaaS: 952013314444
- For Amazon FSx for NetApp ONTAP specifically, edit the **Trust relationships** policy to include "AWS": "arn:aws:iam::952013314444:root".

For example, the policy should look like this:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::952013314444:root",
        "Service": "ec2.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Refer to [AWS Identity and Access Management \(IAM\) documentation](#) for more information on cross account resource access in IAM.

- Create a policy that includes the permissions required to create a Console agent.
 - [View the permissions needed for FSx for ONTAP](#)
 - [View the agent deployment policy](#)

3. Copy the Role ARN of the IAM role so that you can paste it in the Console in the next step.

Result

The IAM role now has the required permissions. [You can now add it to the Console.](#)

Add the credentials

After you provide the IAM role with the required permissions, add the role ARN to the Console.

Before you begin

If you just created the IAM role, it might take a few minutes until they are available for use. Wait a few minutes before you add the credentials to the Console.

Steps

1. Select **Administration > Credentials**.



2. On the **Organization credentials** page, select **Add Credentials** and follow the steps in the wizard.
 - a. **Credentials Location:** Select **Amazon Web Services > Console**.
 - b. **Define Credentials:** Provide the ARN (Amazon Resource Name) of the IAM role.
 - c. **Review:** Confirm the details about the new credentials and select **Add**.

Add credentials to the Console for Amazon FSx for ONTAP

For details, refer to the [the Console documentation for Amazon FSx for ONTAP](#)

Configure an AWS subscription

After you add your AWS credentials, you can configure an AWS Marketplace subscription with those credentials. The subscription enables you to pay for NetApp data services and Cloud Volumes ONTAP at an hourly rate (PAYGO) or using an annual contract.

There are two scenarios in which you might configure an AWS Marketplace subscription after you've already added the credentials:

- You didn't configure a subscription when you initially added the credentials.
- You want to change the AWS Marketplace subscription that is configured to the AWS credentials.

Replacing the current marketplace subscription with a new subscription changes the marketplace subscription for any existing Cloud Volumes ONTAP systems and all new systems.

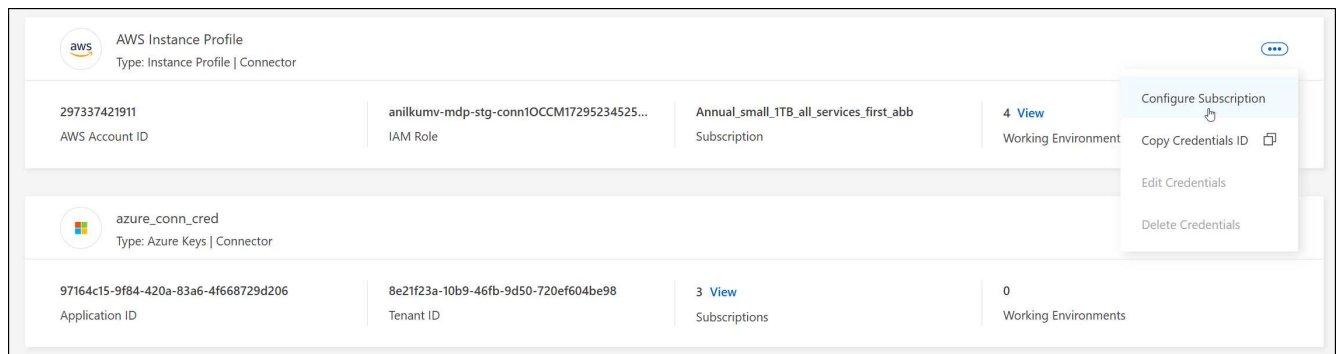
Before you begin

You need to create a Console agent before you can configure a subscription. [Learn how to create a Console agent](#).

Steps

1. Select **Administration > Credentials**.
2. Select **Organization credentials**.
3. Select the action menu for a set of credentials that are associated with a Console agent and then select **Configure Subscription**.

You must select credentials that are associated with a Console agent. You can't associate a marketplace subscription with credentials that are associated with the NetApp Console.



4. To associate the credentials with an existing subscription, select the subscription from the down-down list and select **Configure**.
5. To associate the credentials with a new subscription, select **Add Subscription > Continue** and follow the steps in the AWS Marketplace:
 - a. Select **View purchase options**.
 - b. Select **Subscribe**.
 - c. Select **Set up your account**.

You'll be redirected to the NetApp Console.

d. From the **Subscription Assignment** page:

- Select the Console organizations or accounts that you'd like to associate this subscription with.
- In the **Replace existing subscription** field, choose whether you'd like to automatically replace the existing subscription for one organization or account with this new subscription.

The Console replaces the existing subscription for all credentials in the organization or account with this new subscription. If a set of credentials wasn't ever associated with a subscription, then this new subscription won't be associated with those credentials.

For all other organizations or accounts, you'll need to manually associate the subscription by repeating these steps.

- Select **Save**.

Associate an existing subscription with your organization

When you subscribe to from the AWS Marketplace, the last step in the process is to associate the subscription with your organization. If you didn't complete this step, then you can't use the subscription with your organization.

- [Learn about the Console deployment modes](#)
- [Learn about the Console identity and access management](#)

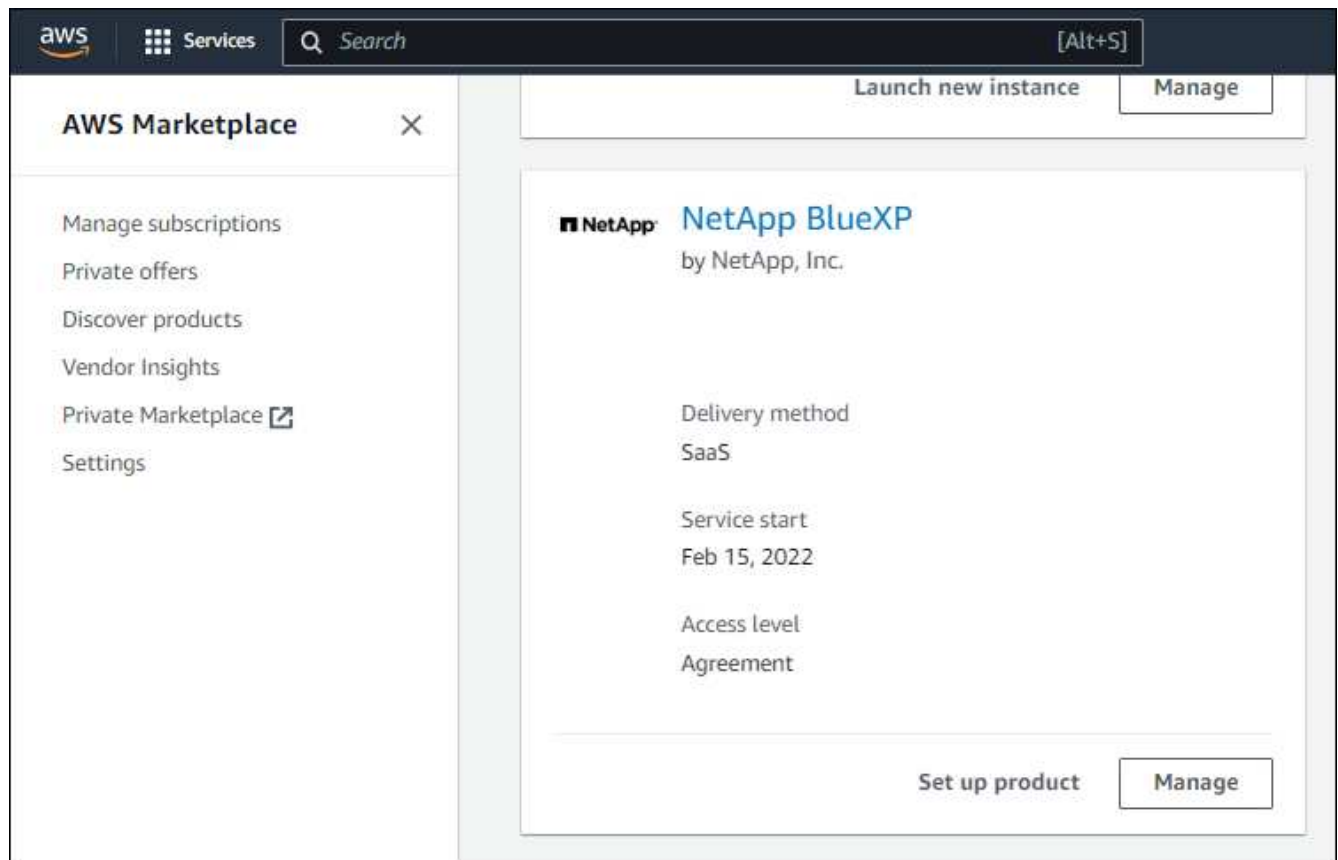
Follow the steps below if you subscribed to NetApp Intelligent Services from the AWS Marketplace, but you missed the step to associate the subscription with your account.

Steps

1. Confirm that you didn't associate your subscription with your Console organization.
 - a. From the navigation menu, select **Administration > Licenses and subscriptions**.
 - b. Select **Subscriptions**.
 - c. Verify that your subscription doesn't appear.

You'll only see the subscriptions that are associated with the organization or account that you're currently viewing. If you don't see your subscription, proceed with the following steps.

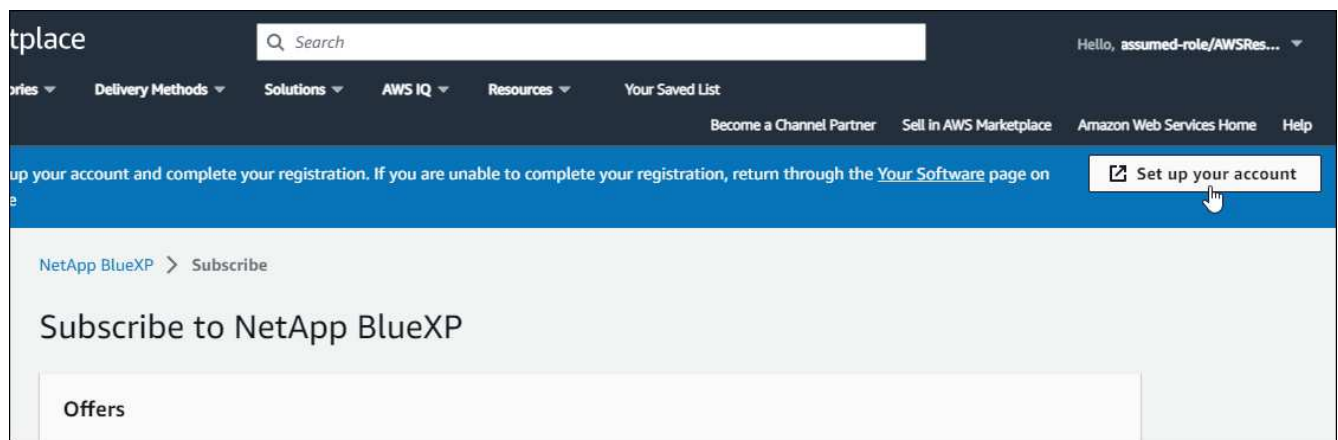
2. Log in to the AWS Console and navigate to **AWS Marketplace Subscriptions**.
3. Find the subscription.



4. Select **Set up product**.

The subscription offer page should load in a new browser tab or window.

5. Select **Set up your account**.



The **Subscription Assignment** page on netapp.com should load in a new browser tab or window.

Note that you might be prompted to log in to the Console first.

6. From the **Subscription Assignment** page:
 - Select the Console organizations or accounts that you'd like to associate this subscription with.
 - In the **Replace existing subscription** field, choose whether you'd like to automatically replace the existing subscription for one organization or account with this new subscription.

The Console replaces the existing subscription for all credentials in the organization or account with this new subscription. If a set of credentials wasn't ever associated with a subscription, then this new subscription won't be associated with those credentials.

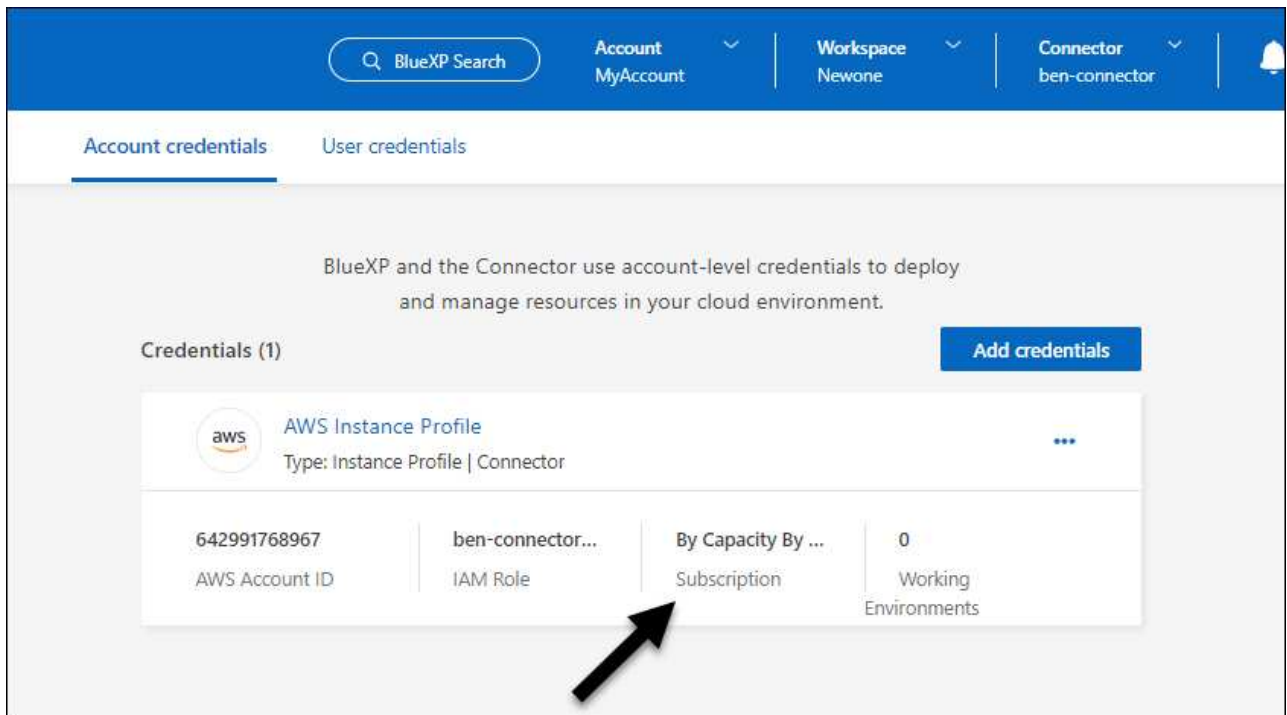
For all other organizations or accounts, you'll need to manually associate the subscription by repeating these steps.

The screenshot shows a 'Subscription Assignment' dialog box. At the top, there is a success message: 'Your subscription to BlueXP / Cloud Volumes ONTAP from the AWS Marketplace was created successfully.' Below this, the 'Subscription name' is set to 'PayAsYouGo'. A section titled 'Select the NetApp accounts that you'd like to associate this subscription with.' contains a table with three rows. The first two rows have their 'Replace existing subscription' toggle switches turned off, while the third row, for 'benAccount', has its toggle switch turned on. A 'Save' button is located at the bottom right of the dialog.

NetApp account	Replace existing subscription
<input checked="" type="checkbox"/> cloudTiering_undefined	<input type="checkbox"/>
<input checked="" type="checkbox"/> CS-HhewH	<input type="checkbox"/>
<input checked="" type="checkbox"/> benAccount	<input checked="" type="checkbox"/>

7. Confirm that the subscription is associated with your organization.
 - a. From the navigation menu, select **Administration > License and subscriptions**.
 - b. Select **Subscriptions**.
 - c. Verify that your subscription appears.
8. Confirm that the subscription is associated with your AWS credentials.
 - a. Select **Administration > Credentials**.
 - b. On the **Organization credentials** page, verify that the subscription is associated with your AWS credentials.

Here's an example.



Edit credentials

Edit your AWS credentials by changing the account type (AWS keys or assume role), by editing the name, or by updating the credentials themselves (the keys or the role ARN).



You can't edit the credentials for an instance profile that are associated with a Console agent instance or an Amazon FSx for ONTAP instance. You can only rename the credentials for an FSx for ONTAP instance.

Steps

1. Select **Administration > Credentials**.
2. On the **Organization credentials** page, select the action menu for a set of credentials and then select **Edit Credentials**.
3. Make the required changes and then select **Apply**.

Delete credentials

If you no longer need a set of credentials, you can delete them. You can only delete credentials that aren't associated with a system.



You can't delete the credentials for an instance profile that is associated with a Console agent.

Steps

1. Select **Administration > Credentials**.
2. On the **Organization credentials** or **Account credentials** page, select the action menu for a set of credentials and then select **Delete Credentials**.
3. Select **Delete** to confirm.

Azure

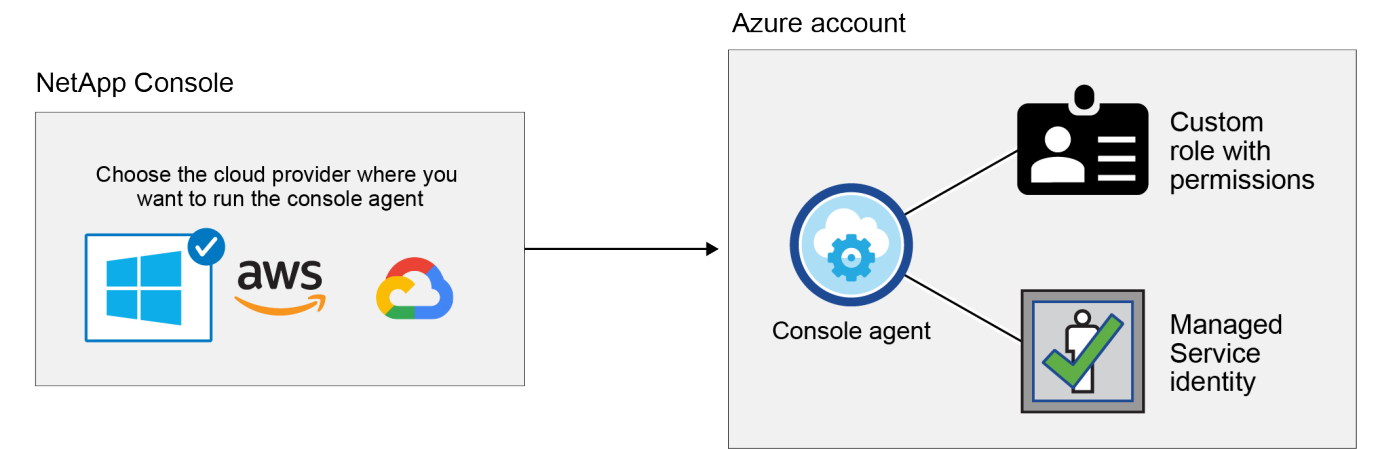
Learn about Azure credentials and permissions in NetApp Console

Learn how the NetApp Console uses Azure credentials to perform actions on your behalf and how those credentials are associated with marketplace subscriptions. Understanding these details can be helpful as you manage the credentials for one or more Azure subscriptions. For example, you might want to learn when to add additional Azure credentials to the Console.

Initial Azure credentials

When you deploy a Console agent from the Console, you need to use an Azure account or service principal that has permissions to deploy the Console agent virtual machine. The required permissions are listed in the [Agent deployment policy for Azure](#).

When the Console deploys the Console agent virtual machine in Azure, it enables a [system-assigned managed identity](#) on the virtual machine, creates a custom role, and assigns it to the virtual machine. The role provides the Console with the permissions required to manage resources and processes within that Azure subscription. [Review how the Console uses the permissions](#).



If you create a new system for Cloud Volumes ONTAP, the Console selects these Azure credentials by default:

Details & Credentials			
Managed Service Ide...	OCCM QA1	No subscription is associated	
Credential Name	Azure Subscription	Marketplace Subscription	Edit Credentials

You can deploy all of your Cloud Volumes ONTAP systems using the initial Azure credentials, or you can add additional credentials.

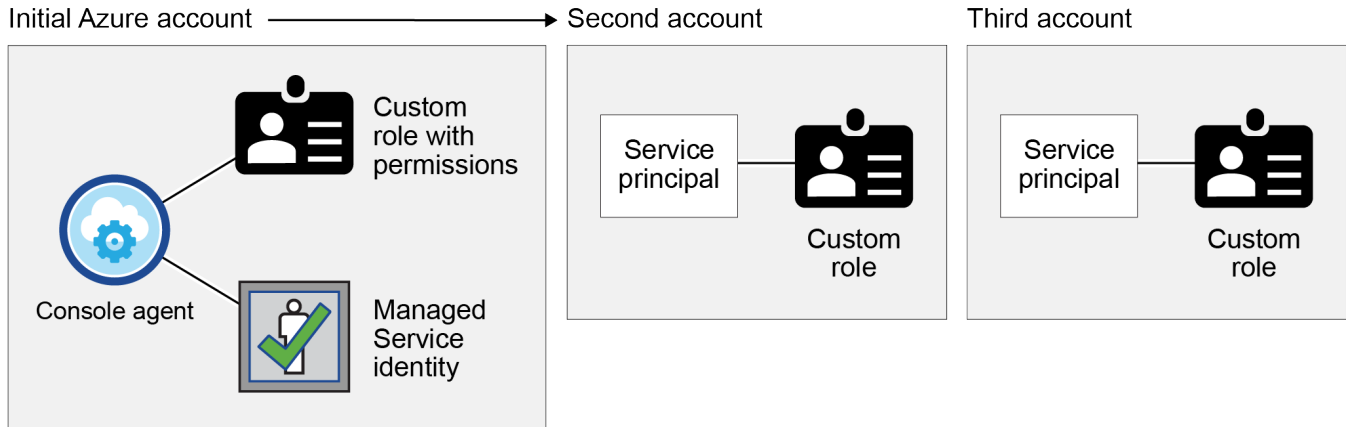
Additional Azure subscriptions for a managed identity

The system-assigned managed identity assigned to the Console agent VM is associated with the subscription in which you launched the Console agent. If you want to select a different Azure subscription, then you need to

associate the managed identity with those subscriptions.

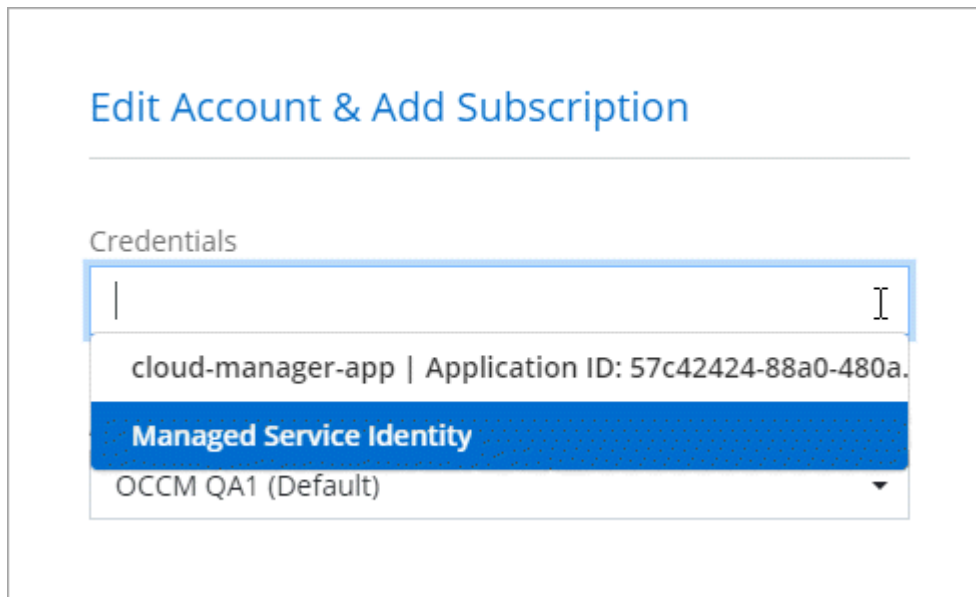
Additional Azure credentials

If you want to use different Azure credentials with the Console, then you must grant the required permissions by [creating and setting up a service principal in Microsoft Entra ID](#) for each Azure account. The following image shows two additional accounts, each set up with a service principal and custom role that provides permissions:



You would then [add the account credentials to the Console](#) by providing details about the AD service principal.

For example, you can switch between credentials when creating a new Cloud Volumes ONTAP system:



Credentials and marketplace subscriptions

The credentials that you add to a console agent must be associated with an Azure Marketplace subscription so that you can pay for Cloud Volumes ONTAP at an hourly rate (PAYGO) or NetApp data services or through an annual contract.

[Learn how to associate an Azure subscription.](#)

Note the following about Azure credentials and marketplace subscriptions:

- You can associate only one Azure Marketplace subscription with a set of Azure credentials
- You can replace an existing marketplace subscription with a new subscription

FAQ

The following question is related to credentials and subscriptions.

Can I change the Azure Marketplace subscription for Cloud Volumes ONTAP systems?

Yes, you can. When you change the Azure Marketplace subscription that's associated with a set of Azure credentials, all existing and new Cloud Volumes ONTAP systems will be charged against the new subscription.

[Learn how to associate an Azure subscription.](#)

Can I add multiple Azure credentials, each with different marketplace subscriptions?

All Azure credentials that belong to the same Azure subscription will be associated with the same Azure Marketplace subscription.

If you have multiple Azure credentials that belong to different Azure subscriptions, then those credentials can be associated with the same Azure Marketplace subscription or with different marketplace subscriptions.

Can I move existing Cloud Volumes ONTAP systems to a different Azure subscription?

No, it's not possible to move the Azure resources associated with your Cloud Volumes ONTAP system to a different Azure subscription.

How do credentials work for marketplace deployments and on-premises deployments?

The sections above describe the recommended deployment method for the Console agent, which is from the Console. You can also deploy a console agent in Azure from the Azure Marketplace, and you can install the Console agent software on your own Linux host.

If you use the Marketplace, you can provide permissions by assigning a custom role to the Console agent VM and to a system-assigned managed identity, or you can use a Microsoft Entra service principal.

For on-premises deployments, you can't set up a managed identity for the Console agent, but you can provide permissions by using a service principal.

To learn how to set up permissions, refer to the following pages:

- Standard mode
 - [Set up permissions for an Azure Marketplace deployment](#)
 - [Set up permissions for on-premises deployments](#)
- Restricted mode
 - [Set up permissions for restricted mode](#)

Manage Azure credentials and marketplace subscriptions for NetApp Console

Add and manage Azure credentials so that the NetApp Console has the permissions that it needs to deploy and manage cloud resources in your Azure subscriptions. If you manage multiple Azure Marketplace subscriptions, you can assign each one of them to

different Azure credentials from the Credentials page.

Overview

There are two ways to add additional Azure subscriptions and credentials in the Console.

1. Associate additional Azure subscriptions with the Azure managed identity.
2. To deploy Cloud Volumes ONTAP using different Azure credentials, grant Azure permissions using a service principal and add its credentials to the Console.

Associate additional Azure subscriptions with a managed identity

The Console enables you to choose the Azure credentials and Azure subscription in which you want to deploy Cloud Volumes ONTAP. You can't select a different Azure subscription for the managed identity profile unless you associate the [managed identity](#) with those subscriptions.

About this task

A managed identity is [the initial Azure account](#) when you deploy a Console agent from the Console. When you deploy the Console agent, the Console assigns the Console Operator role to the Console agent virtual machine.

Steps

1. Log in to the Azure portal.
2. Open the **Subscriptions** service and then select the subscription in which you want to deploy Cloud Volumes ONTAP.
3. Select **Access control (IAM)**.
 - a. Select **Add > Add role assignment** and then add the permissions:
 - Select the **Console Operator** role.



Console Operator is the default name provided in a Console agent policy. If you chose a different name for the role, then select that name instead.

- Assign access to a **Virtual Machine**.
 - Select the subscription in which a Console agent virtual machine was created.
 - Select a Console agent virtual machine.
 - Select **Save**.
4. Repeat these steps for additional subscriptions.

Result

When creating a new system, you can now select from multiple Azure subscriptions for the managed identity profile.

Edit Account & Add Subscription

Credentials

Managed Service Identity

Azure Subscription

OCCM Dev

OCCM QA1 (Default)

No subscription is associated with this account

Add additional Azure credentials to NetApp Console

When you deploy a Console agent from the Console, the Console enables a system-assigned managed identity on the virtual machine that has the required permissions. The Console selects these Azure credentials by default when you create a new system for Cloud Volumes ONTAP.



An initial set of credentials isn't added if you manually installed a Console agent software on an existing system. [Learn about Azure credentials and permissions.](#)

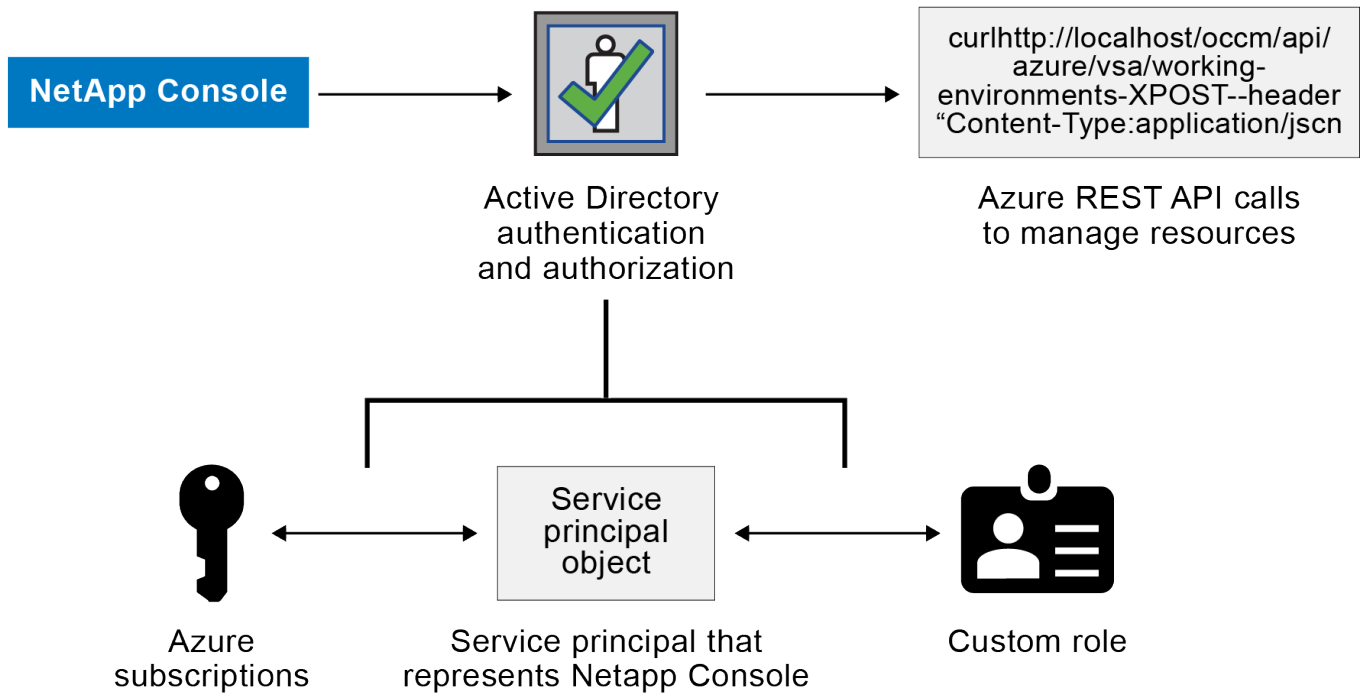
If you want to deploy Cloud Volumes ONTAP using *different* Azure credentials, then you must grant the required permissions by creating and setting up a service principal in Microsoft Entra ID for each Azure account. You can then add the new credentials to the Console.

Grant Azure permissions using a service principal

The Console needs permissions to perform actions in Azure. You can grant the required permissions to an Azure account by creating and setting up a service principal in Microsoft Entra ID and by obtaining the Azure credentials that the Console needs.

About this task

The following image depicts how the Console obtains permissions to perform operations in Azure. A service principal object, which is tied to one or more Azure subscriptions, represents the Console in Microsoft Entra ID and is assigned to a custom role that allows the required permissions.



Steps

1. [Create a Microsoft Entra 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 a Microsoft Entra application

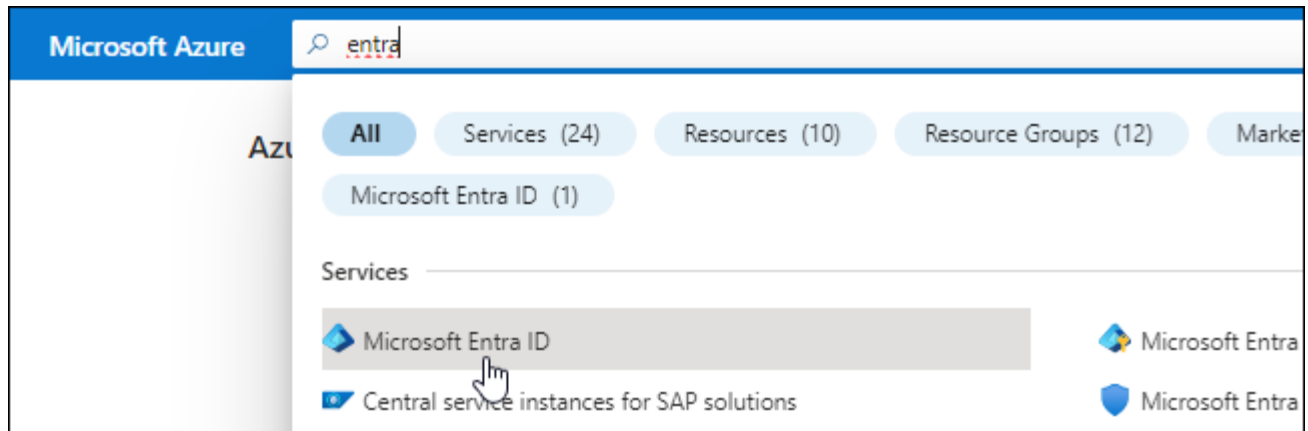
Create a Microsoft Entra application and service principal that the Console can use for role-based access control.

Steps

1. Ensure that you have 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](#)

2. From the Azure portal, open the **Microsoft Entra ID** service.



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

You've created the AD application and service principal.

Assign the application to a role

You must bind the service principal to one or more Azure subscriptions and assign it the custom "Console Operator" role so the Console has permissions in Azure.

Steps

1. Create a custom role:

Note that you can create an Azure custom role using the Azure portal, Azure PowerShell, Azure CLI, or REST API. The following steps show how to create the role using the Azure CLI. If you would prefer to use a different method, refer to [Azure documentation](#)

- a. Copy the contents of the [custom role permissions for the Console agent](#) and save them in a JSON file.
- 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.

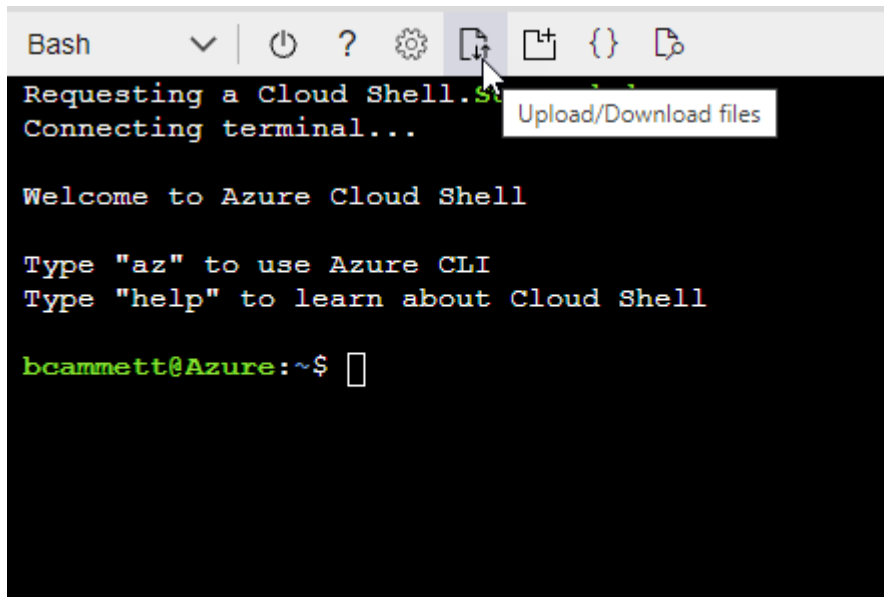
Example

```
"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 steps describe how to create the role by using Bash in Azure Cloud Shell.

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



- Use the Azure CLI to create the custom role:

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

You should now have a custom role called Console Operator that you can assign to the Console agent virtual machine.

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

Add role assignment ...

Got feedback?

Role **Members** Review + assign

Selected role Cloud Manager Operator 3.9.12_B

Assign access to ☒ User, group, or service principal ☐ Managed identity

Members + [Select members](#)

- Search for the name of the application.

Here's an example:

Select members ✕

Select ⓘ

test-service-principal

test-service-principal

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

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

If you want to deploy Cloud Volumes ONTAP from multiple Azure subscriptions, then you must bind the service principal to each of those subscriptions. In the NetApp Console, you can select the subscription that you want to use when deploying Cloud Volumes ONTAP.

Add Windows Azure Service Management API permissions

You must assign "Windows Azure Service Management API" permissions to the service principal.

Steps


1. In the **Microsoft Entra ID** service, select **App registrations** and select the application.
2. Select **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.

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

4. Select **Access Azure Service Management as organization users** and then select **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](#)

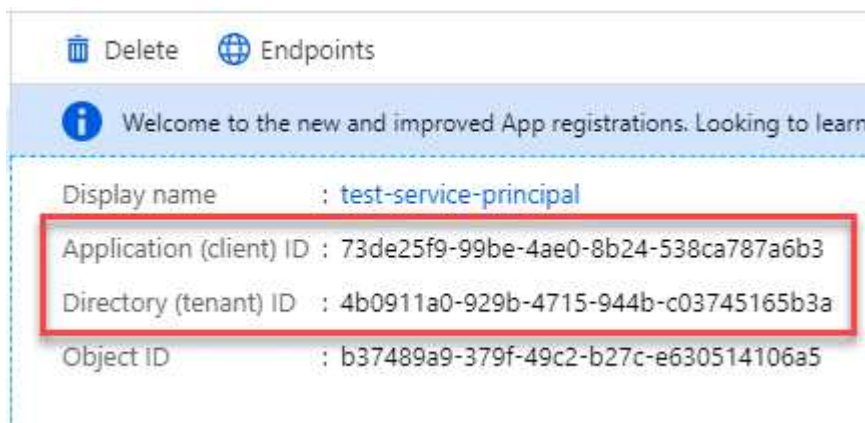
Type to search	
PERMISSION	ADMIN CONSENT REQUIRED
<input checked="" type="checkbox"/> user_impersonation Access Azure Service Management as organization users (preview) ⓘ	-

Get the application ID and directory ID

When you add the Azure account to the Console, you need to provide the application (client) ID and the directory (tenant) ID for the application. The Console uses the IDs to programmatically sign in.

Steps

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



When you add the Azure account to the Console, you need to provide the application (client) ID and the directory (tenant) ID for the application. The Console uses the IDs to programmatically sign in.

Create a client secret

Create a client secret and provide its value to the Console for authentication with Microsoft Entra ID.

Steps

1. Open the **Microsoft Entra ID** service.

2. Select **App registrations** and select your application.
3. Select **Certificates & secrets > New client secret**.
4. Provide a description of the secret and a duration.
5. Select **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 the Console when you add an Azure account.

Add the credentials to the Console

After you provide an Azure account with the required permissions, you can add the credentials for that account to the Console. Completing this step enables you to launch Cloud Volumes ONTAP using different Azure credentials.

Before you begin

If you just created these credentials in your cloud provider, it might take a few minutes until they are available for use. Wait a few minutes before you add the credentials to the Console.

Before you begin

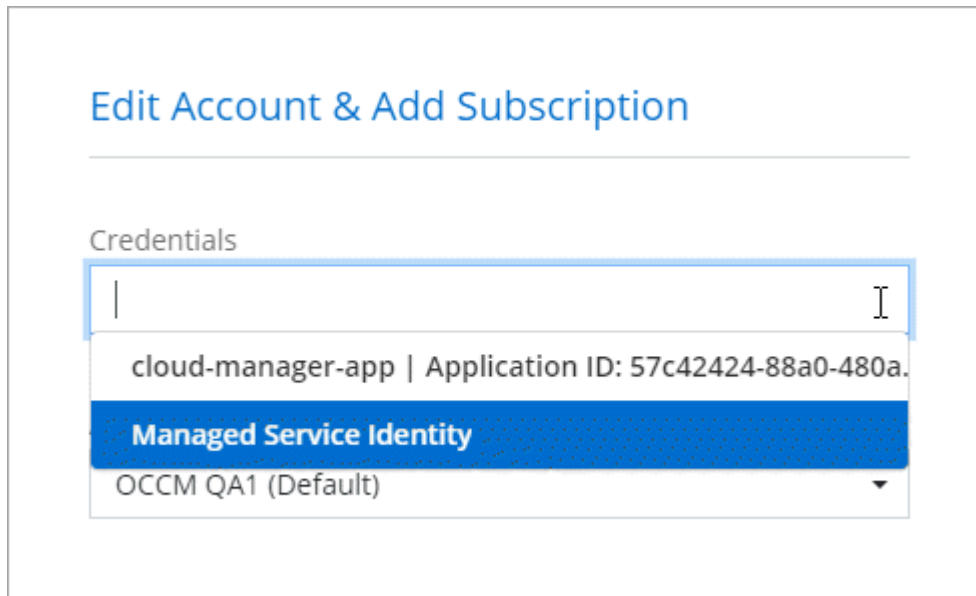
You need to create a Console agent before you can change Console settings. [Learn how to create a Console agent](#).

Steps

1. Select **Administration > Credentials**.
2. Select **Add Credentials** and follow the steps in the wizard.
 - a. **Credentials Location:** Select **Microsoft Azure > Agent**.
 - b. **Define Credentials:** Enter information about the Microsoft Entra service principal that grants the required permissions:
 - Application (client) ID
 - Directory (tenant) ID
 - Client Secret
 - c. **Marketplace Subscription:** Associate a Marketplace subscription with these credentials by subscribing now or by selecting an existing subscription.
 - d. **Review:** Confirm the details about the new credentials and select **Add**.

Result

You can switch to a different set of credentials from the Details and Credentials page [when adding a system to the Console](#)



Manage existing credentials

Manage the Azure credentials that you've already added to the Console by associating a Marketplace subscription, editing credentials, and deleting them.

Associate an Azure Marketplace subscription to credentials

After you add your Azure credentials to the Console, you can associate an Azure Marketplace subscription to those credentials. You can use the subscription to create a pay-as-you-go Cloud Volumes ONTAP system and access NetApp data services.

There are two scenarios in which you might associate an Azure Marketplace subscription after you've already added the credentials to the Console:

- You didn't associate a subscription when you initially added the credentials to the Console.
- You want to change the Azure Marketplace subscription that is associated with Azure credentials.

Replacing the current marketplace subscription updates it for existing and new Cloud Volumes ONTAP systems.

Steps

1. Select **Administration > Credentials**.
2. Select **Organization credentials**.
3. Select the action menu for a set of credentials that are associated with a Console agent and then select **Configure Subscription**.

You must select credentials that are associated with a Console agent. You can't associate a marketplace subscription with credentials that are associated with the NetApp Console.

4. To associate the credentials with an existing subscription, select the subscription from the down-down list

and select **Configure**.

5. To associate the credentials with a new subscription, select **Add Subscription > Continue** and follow the steps in the Azure Marketplace:
 - a. If prompted, log in to your Azure account.
 - b. Select **Subscribe**.
 - c. Fill out the form and select **Subscribe**.
 - d. After the subscription process is complete, select **Configure account now**.

You'll be redirected to the NetApp Console.

- e. From the **Subscription Assignment** page:
 - Select the Console organizations or accounts that you'd like to associate this subscription with.
 - In the **Replace existing subscription** field, choose whether you'd like to automatically replace the existing subscription for one organization or account with this new subscription.

The Console replaces the existing subscription for all credentials in the organization or account with this new subscription. If a set of credentials wasn't ever associated with a subscription, then this new subscription won't be associated with those credentials.

For all other organizations or accounts, you'll need to manually associate the subscription by repeating these steps.

- Select **Save**.

Edit credentials

Edit your Azure credentials in the Console. For example, you can update the client secret if a new secret was created for the service principal application.

Steps

1. Select **Administration > Credentials**.
2. Select **Organization credentials**.
3. Select the action menu for a set of credentials and then select **Edit Credentials**.
4. Make the required changes and then select **Apply**.

Delete credentials

If you no longer need a set of credentials, you can delete them. You can only delete credentials that aren't associated with a system.

Steps

1. Select **Administration > Credentials**.
2. Select **Organization credentials**.
3. On the **Organization credentials** page, select the action menu for a set of credentials and then select **Delete Credentials**.
4. Select **Delete** to confirm.

Google Cloud

Learn about Google Cloud projects and permissions

Learn how the NetApp Console uses Google Cloud credentials to perform actions on your behalf and how those credentials are associated with marketplace subscriptions. Understanding these details can be helpful as you manage the credentials for one or more Google Cloud projects. For example, you might want to learn about the service account that's associated with the Console agent VM.

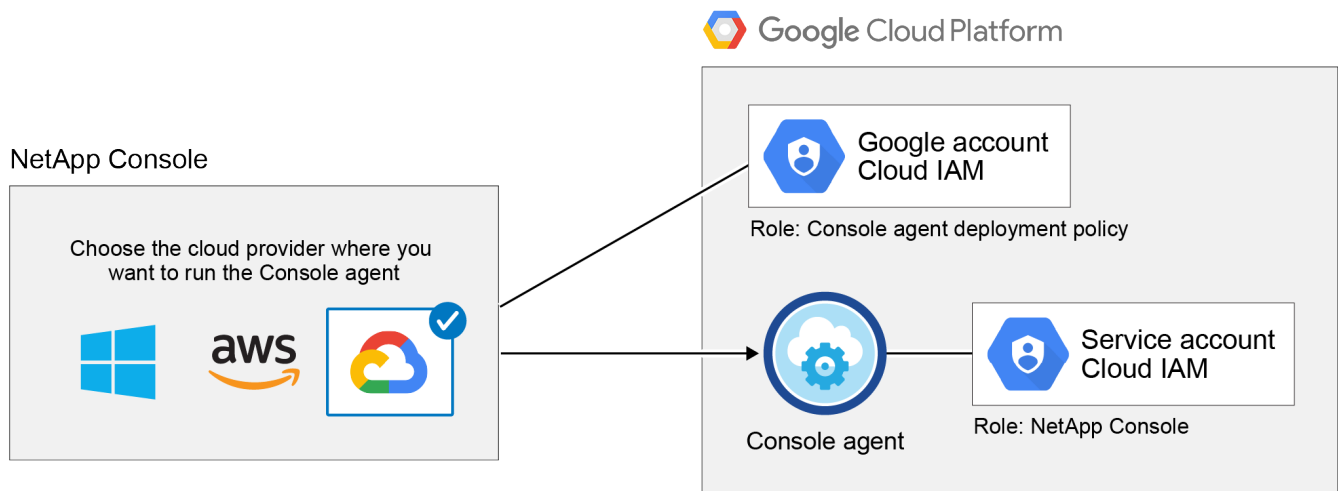
Project and permissions for NetApp Console

Before you can use the Console to manage resources in your Google Cloud project, you must first deploy a Console agent. The agent can't be running on your premises, or in a different cloud provider.

Two sets of permissions must be in place before you deploy a Console agent directly from the Console:

1. You need to deploy a Console agent using a Google account that has permissions to launch the Console agent from the Console.
2. When deploying the Console agent, you are prompted to select a [service account](#) for the agent. The Console gets permissions from the service account to create and manage Cloud Volumes ONTAP systems, to manage backups using NetApp backup and recovery, and more. Permissions are provided by attaching a custom role to the service account.

The following image depicts the permission requirements described in numbers 1 and 2 above:



To learn how to set up permissions, refer to the following pages:

- [Set up Google Cloud permissions for standard mode](#)
- [Set up permissions for restricted mode](#)

Credentials and marketplace subscriptions

When you deploy a Console agent in Google Cloud, the Console creates a default set of credentials for the Google Cloud service account in the project in which the Console agent resides. These credentials must be associated with a Google Cloud Marketplace subscription so that you can pay for Cloud Volumes ONTAP and NetApp data services.

[Learn how to associate a Google Cloud Marketplace subscription.](#)

Note the following about Google Cloud credentials and marketplace subscriptions:

- Only one set of Google Cloud credentials can be associated with a Console agent
- You can associate only one Google Cloud Marketplace subscription with the credentials
- You can replace an existing marketplace subscription with a new subscription

Project for Cloud Volumes ONTAP

Cloud Volumes ONTAP can reside in the same project as the Console agent, or in a different project. To deploy Cloud Volumes ONTAP in a different project, you need to first add the Console agent service account and role to that project.

- [Learn how to set up the service account](#)
- [Learn how to deploy Cloud Volumes ONTAP in Google Cloud and select a project](#)

Manage Console agent permissions for Google Cloud deployments

Occasionally, NetApp updates the permissions required for the service account used for the Console agent when it is deployed in Google Cloud.

[Verify the required Google permissions list.](#)

Use Google Cloud Console to update the IAM role assigned to the service account to match the new set of permissions.

[Google Cloud docs: Edit a custom role](#)

Identity and access management

Learn about NetApp Console identity and access management

Use NetApp Console's Identity and Access Management (IAM) to organize your NetApp resources and control access according to your business structure—by location, department, or project.

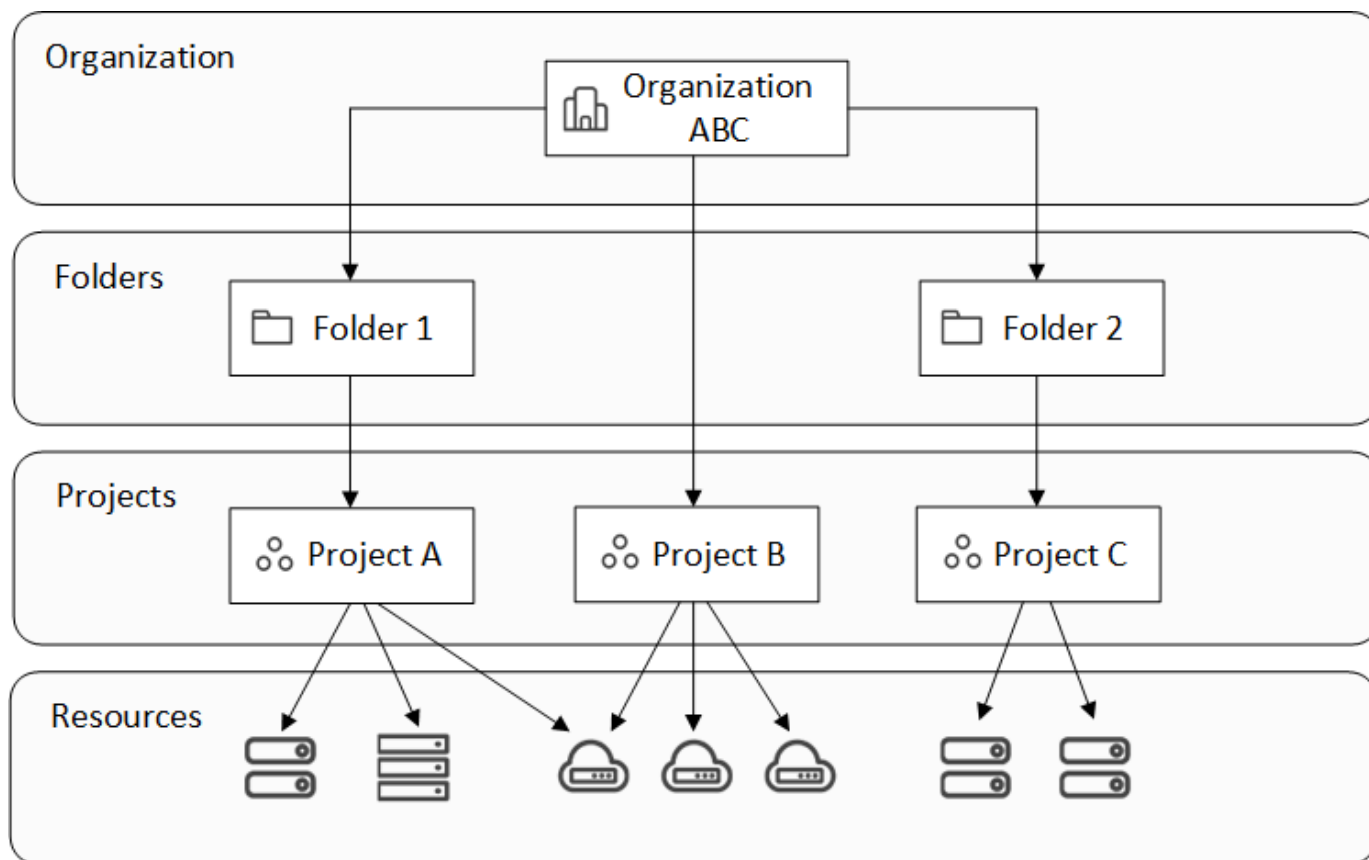
Resources are arranged hierarchically: the organization is at the top, followed by folders (which can contain other folders or projects), and then projects, which contain storage systems, workloads, and agents.

Assign role-based access control (RBAC) permissions to members at the organization, folder, or project level to ensure users have the appropriate access to resources.



You must have the *Super admin*, *Organization admin*, or *Folder or project admin* roles to manage IAM in NetApp Console.

The following image illustrates this hierarchy at a basic level.



Identity and access management components

Within NetApp Console, you organize your storage resources using three main components: organizational components, resource components, and user access components.

Projects and folders within your organization

Within your IAM structure, you work with three organizational components: organizations, projects, and folders. You can grant users access by assigning them roles at any of these levels.

Organization

An *organization* is the top level of the Console IAM system and typically represents your company. Your organization consists of folders, projects, members, roles, and resources. Agents are associated with specific projects in the organization.

Projects

A *project* is used to provide access to a storage resource. You must assign a resource to a project before anyone can access them. You can assign multiple resources to a single project and you can also have multiple projects. You then assign users permissions to the project to give them access to the resources within it.

For example, you can associate an on-premises ONTAP system with a single project or with all projects in your organization, depending on your needs.

[Learn how to add projects to your organization.](#)

Folders

Group related projects in *folders* to organize them by location, site, or business unit. You can't associate resources directly with folders, but assigning a user a role at the folder level gives them access to all projects in that folder.

[Learn how to add folders to your organization.](#)

Resources

Resources include storage systems, Keystone subscriptions, as well as Console agents.

+

You must associate a resource with a project before anyone can access it.

+

For example, you might associate a Cloud Volumes ONTAP system with one project or with all projects in your organization. How you associate a resource depends on your organization's needs.

+

[Learn how to associate resources to projects.](#)

Storage systems and Keystone subscriptions

Storage systems are the primary resources that you manage in NetApp Console. NetApp Console supports management of both on-premises and cloud storage systems. You must add a storage system to a project before anyone can access it.

Storage systems are automatically associated with the project where they are added, but you can also associate them with other projects or folders from the **Resources** page.

Keystone subscriptions are also resources that you can associate with projects in order to grant users access to the subscription in NetApp Console.

Console agents

Organization admins create Console agents to manage storage systems and enable NetApp data services. Agents are initially tied to the project where they are created, but admins can add them to other projects or folders from the Agents page.

Associating an agent with a project enables management of resources in that project, while associating an agent with a folder lets folder or project admins decide which projects should use the agent. Agents must be linked to specific projects to provide management capabilities.

[Learn how to associate agents with projects.](#)

Members and roles

Members

Members of your organization are user accounts or service accounts. A service account is typically used by an application to complete specified tasks without human intervention.

You need to add members to your organization after they sign up for NetApp Console. Once added, you can assign them roles to provide access to resources. You can manually add service accounts from within the Console or automate their creation and management through the NetApp Console IAM API.

[Learn how to add members to your organization.](#)

Access roles

The Console provides access roles that you can assign to the members of your organization.

When you associate a member with a role, you can grant that role for the entire organization, a specific folder, or a specific project. The role that you select gives a member permissions to the resources in the selected part of the hierarchy.

NetApp Console provides granular roles that adhere to the principles of "least privilege" which means access roles are designed to give users access to only that that they need

This means users may have multiple roles assigned to them as their duties expand.

[Learn about access roles.](#)

IAM strategy examples

Small organization strategy

For organizations with fewer than 50 users and centralized storage management, consider a simplified approach using Super admin and Super viewer roles.

Example: ABC Corporation (5-person team)

- **Structure:** Single organization with 3 projects (Production, Development, Backup)
- **Roles:**
 - 2 senior members: **Super admin** role for full administrative access
 - 3 team members: **Super viewer** role for monitoring without modification rights
- **Agent strategy:** Single agent associated with all projects for shared resource access
- **Benefits:** Simplified administration, reduced role complexity, suitable for teams requiring broad access

Multi-regional enterprise strategy

For large organizations with regional operations and specialized teams, implement a hierarchical approach with folders representing geographical or business unit boundaries.

Example: XYZ Corporation (multinational company)

- **Structure:** Organization > Regional folders (North America, Europe, Asia-Pacific) > Project folders per region
- **Platform roles:**
 - 1 **Organization admin:** Global oversight and policy management
 - 3 **Folder or project admins:** Regional control (one per region)
 - 1 **Federation admin:** Corporate identity provider integration
- **Storage roles by region:**
 - 9 **Storage admin:** Discover and manage storage systems in assigned regions
 - 2 **Storage viewer:** Monitor storage resources across regions
 - 1 **System health specialist:** Manage storage health without system modifications

- **Data service roles:**
 - **Backup and Recovery admin:** Per-project based on backup responsibilities
 - **Ransomware Resilience admin:** Security team monitoring across projects
- **Agent strategy:** Regional agents associated with appropriate geographical projects
- **Benefits:** Enhanced security through role segregation, regional autonomy, and compliance with local regulations

Departmental specialization strategy

For organizations with specialized teams requiring specific data service access, use targeted role assignments based on functional responsibilities.

Example: TechCorp (mid-size technology company)

- **Structure:** Organization > Department folders (IT, Security, Development) > Project-specific resources
- **Specialized roles:**
 - Security team: **Ransomware Resilience admin** and **Classification viewer** roles
 - Backup team: **Backup and Recovery super admin** for comprehensive backup operations
 - Development team: **Storage admin** for test environment management
 - Compliance team: **Operation support analyst** for monitoring and support case management
- **Agent strategy:** Agents linked to departmental projects based on resource ownership
- **Benefits:** Tailored access control, improved operational efficiency, and clear accountability for specialized tasks

Next steps with IAM in NetApp Console

- [Get started with IAM in NetApp Console](#)
- [Monitor or audit IAM activity](#)
- [Learn about the API for NetApp Console IAM](#)

Get started with identity and access in NetApp Console

When you sign up for the NetApp Console, you're prompted to create a new organization. The organization includes one member (an Organization admin) and one default project. To set up identity and access management (IAM) to meet your business needs, you'll need to customize your organization's hierarchy, add additional members, add or discover resources, and associate those resources across your hierarchy.

You need the **Org admin** or **Super admin** permissions to manage identity and access for your organization. With **Folder or project admin** permissions, you can manage only the folders and projects you have access to.

Follow these steps to set up a new organization. The order may vary based on your organization's needs.



Edit the default project or add to your organization's hierarchy

Use the default project or create additional projects and folders matching your business hierarchy.

[Learn how to organize your resources with folders and projects.](#)

2

Associate members with your organization

After users sign up for NetApp Console, you must explicitly add them to your Console organization. You also have the option to add service accounts to your organization.

[Learn how to manage members and their permissions.](#)

3

Add or discover resources

Add or discover resources (systems) to the Console. Organization members manage systems from within a project.

Learn how to create or discover resources:

- [Amazon FSx for NetApp ONTAP](#)
- [Azure NetApp Files](#)
- [Cloud Volumes ONTAP](#)
- [E-Series systems](#)
- [On-premises ONTAP clusters](#)
- [StorageGRID](#)

4

Associate resources with additional projects

Adding or discovering a system in the Console automatically associates the resource with the currently selected project. To make that resource available to another project in your organization, associate it with the respective project. If a Console agent is used to manage the resource, associate the Console agent with the respective project.

- [Learn how to manage your organization's resource hierarchy.](#)
- [Learn how to associate a Console agent with a folder or project.](#)

Related information

- [Learn about identity and access management in NetApp Console](#)
- [Learn about the API for identity and access](#)

Set up your Console organization

Add folders and projects to your NetApp Console organization

Add folders and projects to match your business structure. After you create folders and projects, you can associate resources with them and manage member access to those projects.

The Console automatically creates one project for you when you create a new organization. Most organizations have the need for more than one project, as well as folders to keep things organized. [Learn about the resource hierarchy in NetApp Console.](#)

Using folders and projects to organize resources

In NetApp Console, an organization contains folders and projects that help you organize your resources. Folders help you group related projects, and projects help you manage resources and member access.

Folders

Folders help you organize related projects. You can create nested folders to represent different levels of your organization's structure. For example, you might create a top-level folder for each business unit and then create subfolders for different teams within that business unit. You then create projects within folders.

Folders also enable you to manage member access more efficiently using role inheritance. When you assign roles to members at the folder level, they inherit permissions for all child projects and folders.



Folders are an organizational tool and not visible to members who do not have IAM permissions such as the Org admin, Folder or project admin, or Super admin roles. Members access projects, not folders.

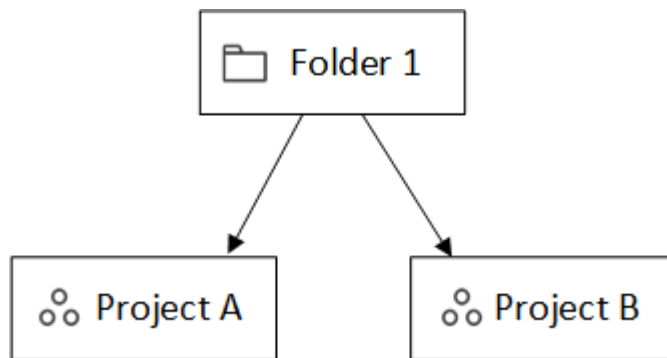
Org admins can delegate administrative responsibilities by creating folders. After creating a folder, an Org admin can assign a member the Folder or project admin roles for particular folders. These members can then manage all projects within that folder without having access to the entire organization.

Folders can have other folders or projects as children, but they cannot have resources directly associated with them. Resources must be associated with a project.

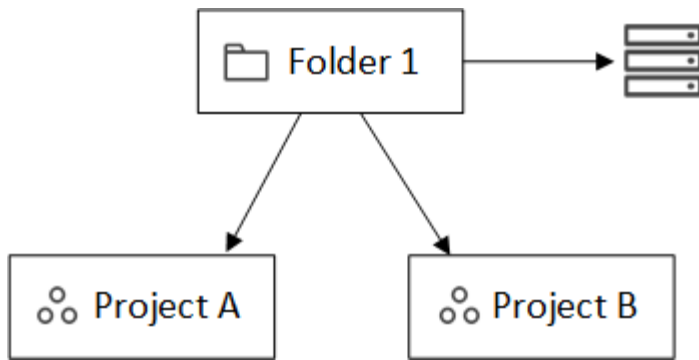
When to associate a resource with a folder

An *Organization administrator* can associate a resource with a folder so a *Folder or project administrator* can link it to the appropriate projects in the folder.

For example, let's say you have a folder that contains two projects:

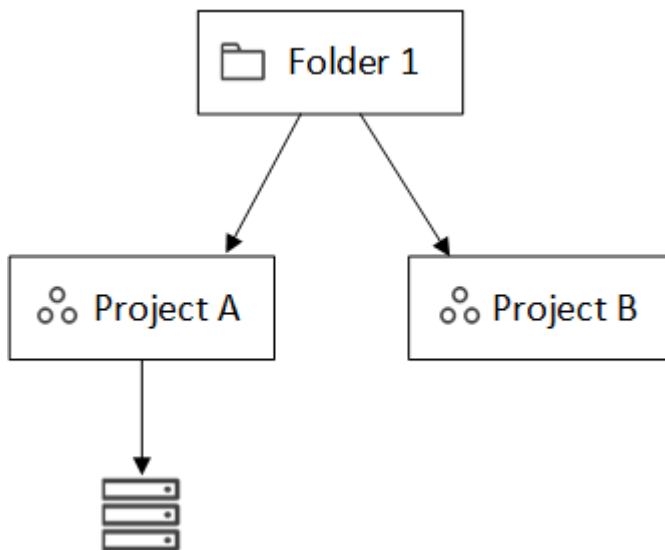


The *Organization admin* can associate a resource with the folder:



Associating a resource with a folder does not make it accessible to all projects; only the *folder or project admin* can see it. The *Folder or project admin* decides which projects can access it and associates the resource with the appropriate projects.

In this example, the admin associates the resource with Project A:



Members who have permissions for project A can now access the resource.

Projects

Associate resources with projects to allow members to manage them. Resources must be associated with a project for management and user access.

An organization can have one or many projects. A project can be directly under the organization or inside a folder. If an agent is used to discover resources within a project, you must also associate the agent with that project.

Users navigate between assigned projects on the **Systems** page to manage the resources associated with each project.

Add a folder or project

Add projects to manage resources and folders to group related projects. When you create a new organization, the Console includes one project.

You can create up to seven levels of folders and projects in your organization's resource structure. Create

nested folders to organize your resources as needed.

Steps

1. Select **Administration > Identity and access**.
2. Select **Organization**.
3. From the **Organization** page, select **Add folder or project**.
4. Select **Folder** or **Project**.
5. Enter folder or project details:
 - **Name and location:** Enter a name and choose a location for the folder or project. You can place folders or projects under the organization or inside another folder.
 - **Resources:** Select the resources that you want to associate with this folder or project. If you haven't added storage systems to the Console yet, you can do this step later.



Members can't access resources in a folder until those resources are assigned to a project. Use folders to hold resources temporarily until you create the necessary projects. This can help the Organization admin delegate resource allocation to a Folder or project admin, who then assigns resources to projects within the folder.

- **Access:** Select **Add a member** to assign access and a role. You can add or remove members from the project or folder at any time.

[Learn about access roles.](#)

6. Select **Add**.

Rename a folder or project

Rename a folder or project as needed. Renaming does not affect associated resources or member access.

Steps

1. From the **Organization** page, navigate to a project or folder in the table, select **...** and then select **Edit folder** or **Edit project**.
2. On the **Edit** page, enter a new name and select **Apply**.

Delete a folder or project

Delete folders and projects you no longer need, such as after team restructuring or project completion.

Before you delete a folder or project, make sure it does not contain any resources. [Learn how to remove resources.](#)

Steps

1. From the **Organization** page, navigate to a project or folder in the table, select **...** and then select **Delete**.
2. Confirm that you want to delete the folder or project.

View the resources associated with a folder or project

View which resources and members are associated with a folder or project.

Steps

1. From the **Organization** page, navigate to a project or folder in the table, select **...** and then select **Edit folder** or **Edit project**.



2. On the **Edit** page, you can view details about the selected folder or project by expanding the **Resources** or **Access** sections.
 - Select **Resources** to view the associated resources. In the table, the **Status** column identifies the resources that are associated with the folder or project.

Available resources (45)				
<input type="checkbox"/>	Platform Type	Resource Type	Resource Name	Status
<input type="checkbox"/>	AWS	Cloud Volumes ONTAP HA	Keystonecvo2	Associated
<input type="checkbox"/>	AWS	Cloud Volumes ONTAP HA	kfuKeystone1vadim	Associated
<input type="checkbox"/>	AWS	Cloud Volumes ONTAP	cvo1Vadim	Associated
<input type="checkbox"/>	AWS	Cloud Volumes ONTAP HA	cvoparts11test	Associated

Change the resources associated with a folder or project

You can change the resources associated with a folder or project as your organization's needs change.

Steps








1. From the **Organization** page, navigate to a project or folder in the table, select **...** and then select **Edit folder** or **Edit project**.
2. On the **Edit** page, select **Resources**.

In the table, the **Status** column identifies the resources that are associated with the folder or project.

3. Select the resources that you'd like to associate or disassociate.
4. Based on the resources that you selected, select either **Associate with the project** or **Disassociate from the project**.

Available resources (45) | Selected (3) Q

Actions: [Associate with the project](#) | [Disassociate from the project](#)

<input type="checkbox"/>	Platform Type	Resource Type	Resource Name	Status
<input checked="" type="checkbox"/>		Cloud Volumes ONTAP HA	Keystonecvo2	Associated
<input checked="" type="checkbox"/>		Cloud Volumes ONTAP HA	kfuKeystone1vadim	Associated
<input checked="" type="checkbox"/>		Cloud Volumes ONTAP	cvo1Vadim	Associated
<input type="checkbox"/>		Cloud Volumes ONTAP HA	cvoparts11test	Associated
<input type="checkbox"/>		Cloud Volumes ONTAP	cvosecondaryparts11	Associated
<input type="checkbox"/>		Cloud Volumes ONTAP HA	keystonetest	Associated
<input type="checkbox"/>		Cloud Volumes ONTAP HA	keystonetesting55	Associated

5. Select **Apply**.

View members associated with a folder or project

You can view the members associated with a folder or project from the **Organization** page.




Steps

- From the **Organization** page, navigate to a project or folder in the table, select **...** and then select **Edit folder** or **Edit project**.
- On the **Edit** page, select **Access** to view the list of members who have access to the selected folder or project.
 - Select **Access** to view the members who have access to the folder or project.

Access ^

Members (2) Q [Learn more about user roles](#) [Add a member](#)

☐ Load users which inherits access

<input type="checkbox"/>	Type	Name	Role	
<input type="checkbox"/>		Gabriel	Folder or project admin	
<input type="checkbox"/>		Ben	Organization admin	

Modify member access to a folder or project

Modify member access to control resource access. Remember that roles assigned at the folder level are inherited by all child projects and folders.

You cannot change member access at lower levels if it is inherited from the folder or organization level. Change the member's permission at the higher hierarchy level to change access. Alternatively, you can [manage permissions from the Members page](#).

Steps

1. From the **Organization** page, navigate to a project or folder in the table, select **...** and then select **Edit folder** or **Edit project**.
2. On the **Edit** page, select **Access** to view the list of members who have access to the selected folder or project.
3. Modify member access:
 - **Add a member:** Select the member that you'd like to add to the folder or project and assign them a role.
 - **Change a member's role:** For any members with a role other than Organization Admin, select their existing role and then choose a new role.
 - **Remove member access:** For members who have a role defined at the folder or project for which you're viewing, you can remove their access.
4. Select **Apply**.

Related information

- [Learn about identity and access in NetApp Console](#)
- [Get started with identity and access](#)
- [Learn about the identity and access API](#)

Add resources to folders and projects in NetApp Console

Control user access to resources by adding them to projects and folders in your NetApp Console organization. Grant access to users at the project level.

A *resource* is an entity that the Console is aware of, such as a storage resource, a Console agent, or a Backup and Recovery workload.

You can view and manage resources from the **Resources** page in the Console.

Console resource types

You can associate several types of resources to projects in your NetApp Console organization:

Storage resources

Storage resources are the most common type of resource in your organization and represent both on-premises and cloud storage systems. When you add a storage system to the Console, you can add it to a folder or project. Until then, the Console marks it as undiscovered and does not display it on the **Resources** page.

Console agents

If you used a Console agent to discover storage systems, add the agent to the same folder or project. This allows users to perform agent-enabled functions, such as data services or Console-native storage management. You can add agents to folders or projects from the **Agents** page in the Console. [Learn how to associate a Console agent with a folder or project.](#)

Keystone subscriptions

If you have Keystone subscriptions in your organization, you can view them on the **Resources** page. You can associate Keystone subscriptions with folders or projects to provide access to members who have permissions for those folders or projects.

View the resources in your organization

You can view both discovered and undiscovered resources associated with your organization. The system finds storage resources and marks them as undiscovered until you add them to the Console.



The Console excludes Amazon FSx for NetApp ONTAP resources from the Resources page because users cannot associate them with a role. You can view these resources on the **Systems** page or from Workloads.

Steps

1. Select **Administration > Identity and access**.
2. Select **Resources**.
3. Select **Advanced Search & Filtering**.
4. Use the available options to find a resource:
 - **Search by resource name:** Enter a text string and select **Add**.
 - **Platform:** Select one or more platforms, such as Amazon Web Services.
 - **Resources:** Select one or more resources, such as Cloud Volumes ONTAP.
 - **Organization, folder, or project:** Select the entire organization, a specific folder, or a specific project.
5. Select **Search**.

Associate a resource with folders and projects

Associate a resource to a folder or project to make it available to members who have permissions for that folder or project.

Steps

1. From the **Resources** page, navigate to a resource in the table, select **...** and then select **Associate to folders or projects**.
2. Select a folder or project and then select **Accept**.
3. To associate an additional folder or project, select **Add folder or project** and then select the folder or project.

Note that you can only select from the folders and projects for which you have admin permissions.

4. Select **Associate resources**.
 - If you associated the resource with projects, members who have permissions for those projects now have the ability to access the resource from the Console.

- If you associated the resource with a folder, a *Folder or project admin* can now access the resource and associate it with a project within the folder. [Learn about associating a resource with a folder.](#)

After you finish

If you discover a resource using a Console agent, associate the Console agent with the project to grant access. Otherwise, the Console agent and its associated resource are not accessible by members without the *Organization admin* role.

[Learn how to associate a Console agent with a folder or project.](#)

View the folders and projects associated with a resource

You can view the folders and projects that are associated with a particular resource.



If you need to find out which organization members have access to the resource, you can [view the members who have access to the folders and projects that are associated with the resource.](#)

Steps

1. From the **Resources** page, navigate to a resource in the table, select **...** and then select **View details**.

The following example shows a resource that is associated with one project.

Folders (0) Project (1)		Associate to folder or project
Type	Associated folders or projects	
	MyOrganization	
	MyOrganization > Project1	



To see which organization members have access to the resource, [view members with access to associated folders and projects.](#)

Remove a resource from a folder or project

To remove a resource from a folder or project, remove its association. This prevents members from managing the resource in that folder or project.



To remove a discovered resource from the entire organization, go to the **Systems** page and remove the system.

Steps

1. From the **Resources** page, navigate to a resource in the table, select **...** and then select **View details**.
2. To remove a resource from a folder or project, select next to the folder or project.
3. Select **Delete** to remove the association.

Related information

- [Learn about identity and access in NetApp Console](#)

- [Get started with identity and access in NetApp Console](#)
- [Learn about the API for identity and access](#)

Associate a Console agent with other folders and projects

Associate Console agents with specific projects to enable resource management and data service access. Resources discovered through a Console agent require both the resource and agent to be associated with the same respective projects for team access.

Super admins and Org admins can create agents and associate any agent with any project or folder. Folder or project admins can only associate existing agents with folders and projects that they have permissions for.

[Learn more about the actions that a *Folder or project admin* can complete.](#)

Steps

1. Select **Administration > Identity and access > Agents**.
2. From the table, find the Console agent that you want to associate.

Use the search above the table to find a specific Console agent or filter the table by resource hierarchy.

3. To view the folders and projects linked to the Console agent, select **...** and then select **View details**.

The page displays details about the folders and projects that are associated with the Console agent.

4. Select **Associate to folder or project**.
5. Select a folder or project and then select **Accept**.
6. To associate the Console agent with an additional folder or project, select **Add a folder or project** and then select the folder or project.
7. Select **Associate Agent**.

After you finish

Associate the Console agent's resources with the same folders and projects from the **Resources** page.

[Learn how to associate a resource with folders and projects.](#)

Related information

- [Learn about NetApp Console agents](#)
- [Learn about NetApp Console identity and access management](#)
- [Get started with identity and access](#)
- [Learn about the API for identity and access management](#)

Add users to your Console organization

Add users to a NetApp Console organization

Within the Console, you grant users access to projects or folders to according to an access role. A *access role* contains a set of permissions that enables a member (user or service account) to perform specific actions at the assigned level of the resource hierarchy.

Required access roles

Super admin, Org admin, or Folder or project admin (for folders and projects that they are administering).
[Learn about access roles.](#)

Understand how access is granted in NetApp Console

NetApp Console uses role-based access control (RBAC) to manage permissions. Assign roles to users individually or through federated groups. Each role defines allowed actions for specific resources.

Note the following about granting access in NetApp Console:

- All users must first sign up for the NetApp Console before they can be granted access to resources
- You must explicitly assign a role to each user in the Console before they can access resources, even if they are members of a federated group that has been assigned a role.
- You can add service accounts directly from the Console and assign them roles.

Add members to your organization

NetApp Console supports three types of members: user accounts, service accounts, and federated groups.

Users must sign up for NetApp Console before you can add them and assign a role, even if they are in a federated group. Create service accounts directly in the Console.

All members must have at least one role explicitly assigned to them in order to access resources.

When adding a member, choose the resource level (organization, folder, or project) and assign a role or roles with the needed permissions.

Add a user

Users sign up for the NetApp Console, but an Org admin or Folder or project admin must add them to an organization, folder, or project so they can access resources.

Before you begin:

The user must have already signed up for the NetApp Console. If they haven't signed up yet, direct them to [sign up for the NetApp Console](#).



If you are adding a user that is part of a federated group, ensure that the user has already signed up for the NetApp Console and been explicitly assigned a role in the Console. NetApp recommends assigning minimum access role such as Organization viewer.

Steps

1. Select **Administration > Identity and access**.
2. Select **Members**.
3. Select **Add a member**.
4. For **Member Type**, keep **User** selected.
5. For **User's email**, enter the user's email address that is associated with the login that they created.
6. Use the **Select an organization, folder, or project** section to choose the level of your resource hierarchy that the member should have permissions for.

Note the following:

- You can select only the folders and projects for which you have permissions.
- When you select an organization or folder, you grant the member permissions to all its contents.
- You can only assign the **Organization admin** role at the organization level.

7. **Select a category** then select a **Role** that provides the member with permissions for the resources that are associated with the organization, folder, or project that you selected.

[Learn about access roles.](#)

8. To give access to more folders, projects, or roles, select **Add role**, choose the folder, project, or role category, and select a role.
9. Select **Add**.

The Console emails instructions to the user.

Add a service account

Service accounts allow you to automate tasks and securely connect with Console APIs. Choose a client ID and secret for simple setups, or JWT (JSON Web Token) for stronger security in automated or cloud-native environments. Select the method that meets your security requirements.

Before you begin:

For JWT authentication, prepare your public key or certificate.

Steps

1. Select **Administration > Identity and access**.
2. Select **Members**.
3. Select **Add a member**.
4. For **Member Type**, select **Service account**.
5. Enter a name for the service account.
6. To use JWT authentication, select **Use private key JWT authentication** and upload your public RSA key or certificate. Skip if using client ID and secret.

Your X.509 certificate. It must be in PEM, CRT, or CER format.

- a. Set up expiry notifications for your certificate. Choose between seven days or 30 days. Expiry notifications are emailed and shown in the Console to users with the Super admin or Org admin role.
7. Use the **Select an organization, folder, or project** section to choose the level of your resource hierarchy that the member should have permissions for.

Note the following:

- You can only select from the folders and projects for which you have permissions.
 - Selecting an organization or folder grants the member permissions to all its contents.
 - You can only assign the **Organization admin** role at the organization level.
8. Select a **Category** then select a **Role** that gives the member permissions for the resources in the organization, folder, or project you selected.

[Learn about access roles.](#)

9. To give access to more folders, projects, or roles, select **Add role**, choose the folder, project, or role category, and select a role.
10. If you didn't choose to use JWT authentication, download or copy the client ID and client secret.

The Console shows the client secret only once. Copy it securely; you can recreate it later if you lose it.

11. If you chose JWT authentication, download or copy the client ID and JWT audience. The Console displays this information only once and does not allow you to retrieve it later.
12. Select **Close**.

Add a federated group to your organization

You can add a federated group from your identity provider (IdP) to your organization and assign it a role or roles. Members of the federated group inherit the roles that you assign to the group in the Console.

Before you can assign a role to a federated group, ensure the following:

- Set up federation between your IdP and the Console. [Learn how to set up federation](#).
- The group must already exist in your IdP and been assigned app access to the Console.
- Users belonging to the group must have already signed up for the NetApp Console and been explicitly assigned a role in the Console. NetApp recommends assigning minimum access role such as Organization viewer.

Steps

1. Select **Administration > Identity and access**.
2. Select **Members**.
3. Select **Add a member**.
4. For **Member Type**, select **Federated Group**.
5. Select the federation of which the group is a member
6. For **Group name**, enter the exact name of the group in your IdP.
7. Use the **Select an organization, folder, or project** section to choose the level of your resource hierarchy that the member should have permissions for.

Note the following:

- You can only select from the folders and projects for which you have permissions.
 - Selecting an organization or folder grants the member permissions to all its contents.
 - You can only assign the **Organization admin** role at the organization level.
8. Select a **Category** then select a **Role** that gives the member permissions for the resources in the organization, folder, or project you selected.

[Learn about access roles](#).

9. To give access to more folders, projects, or roles, select **Add role**, choose the folder, project, or role category, and select a role.

Related information

- [Learn about identity and access management in NetApp Console](#)
- [Get started with identity and access](#)
- [NetApp Console access roles](#)
- [Learn about the API for identity and access](#)

Manage user access and security

Learn about NetApp Console role-based access control (RBAC)

Manage user access to NetApp Console with role-based access control (RBAC), assigning predefined roles at the organization, folder, or project level. Each role grants specific permissions that define what actions users can perform within their assigned scope.

NetApp designs Console roles with least-privilege, so each role includes only the permissions needed for its tasks. This approach enhances security by limiting access to what each member requires.

After you organize resources into folders and projects, assign organization members a role or roles for specific folders or projects, that allow them to perform only the ir responsibilities.

For example, you can assign a member the Ransomware Resilience admin role for a specific project level, allowing them to perform Ransomware Resilience operations for resources within that project, without granting them broader access to the entire organization. This same user can be granted the role for several projects within your organization.

You can assign users multiple roles for the same scope or different scopes, depending on their responsibilities. For example, a smaller organization might have the same user manage both Ransomware Resilience and Backup and Recovery tasks at the organization level, while a larger organization might have different users assigned to each role at the project level.

Types of Console organization members

There are three types of members in a NetApp Console organization:

* *User accounts*: Individual users who log in to the NetApp Console to manage resources. Users must sign up for the NetApp Console before they can be added to an organization.

* *Service accounts*: Non-human accounts used by applications or services to interact with the NetApp Console via APIs. You can add service accounts directly to your Console organization.

* *Federated groups*: Groups synchronized from your identity provider (IdP) that allow you to manage access for multiple users collectively. Each user within a federated group must have signed up for the NetApp Console and been added to your organization with an access role before they can access resources granted to the group.

[Learn how to add members to your organization.](#)

Predefined roles in NetApp Console

NetApp Console includes predefined roles that you can assign to organization members. Each role includes permissions that specify what actions a member can do within their assigned scope (organization, folder, or project).

NetApp Console roles use least-privilege principles that ensure members have only the permissions needed

for their tasks, and categorizes roles by the type of access they provide:

- Platform roles: Provide Console administration permissions
- Data services roles: Provide permissions for managing specific data services, such as Ransomware Resilience and Backup and Recovery
- Application roles: Provide permissions for managing storage as well as audit Console events and alerts

You can assign multiple roles to a member based on their responsibilities. For example, you might assign a member both the Ransomware Resilience admin role and the Backup and Recovery admin role for a specific project.

[Learn about the predefined roles available in NetApp Console.](#)

Manage member access in NetApp Console

Manage member access in your Console organization. Assign roles to set permissions. Remove members when they leave.

Required access roles

Super admin, Org admin, or Folder or project admin (for folders and projects that they are administering).
Link: [reference-iam-predefined-roles.html](#) [Learn about access roles].

You can assign access roles on a project or folder basis. For example, assign a role to a user for two specific projects or assign the role at the folder level to give a user the Ransomware Resilience admin role for all projects in a folder



Add your folders and projects before assigning users access. [Learn how to add folders and projects.](#)

Understand how access is granted in NetApp Console

NetApp Console uses a role-based access control (RBAC) model to manage user permissions. You can assign predefined roles to members individually or through federated groups. You can add and assign roles to service accounts, as well as federated groups. Each role defines what actions a member can perform at the associated resources.

Note the following about granting access in NetApp Console:

- All users must first sign up for the NetApp Console before they can be granted access to resources.
- You must explicitly assign a role to each user in the Console before they can access resources, even if they are members of a federated group that has been assigned a role.
- You can add service accounts directly from the Console and assign them roles.

Using role inheritance

When you assign a role at the organization, folder, or project level in NetApp Console, that role is automatically inherited by all resources within the selected scope. For example, folder-level roles apply to all contained projects, while project-level roles apply to all resources within that project.

View organization members

To understand which resources and permissions are available to a member, you can view the roles assigned to

the member at different levels of your organization's resource hierarchy. [Learn how to use roles to control access to Console resources.](#)

Steps

- 1. Select **Administration > Identity and access.**
- 2. Select **Members.**

The **Members** table lists the members of your organization.

- 3. From the **Members** page, navigate to a member in the table, select **...** and then select **View details.**

View roles(s) assigned to a member

You can verify which roles they are currently assigned.

If you have the *Folder or project admin* role, the page displays all members in the organization. However, you can only view and manage member permissions for the folders and projects for which you have permissions. [Learn more about the actions that a Folder or project admin can complete.](#)

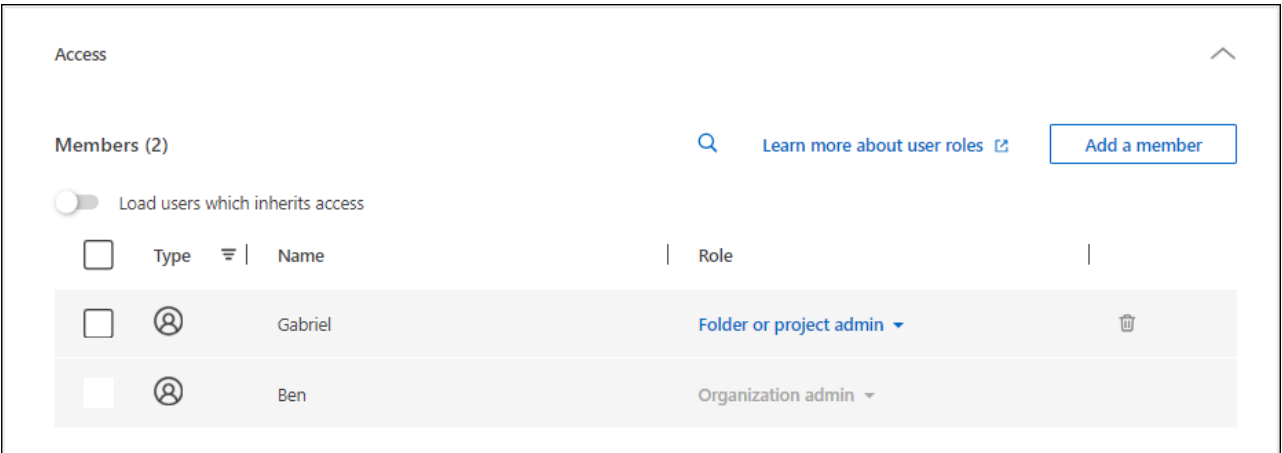
- 1. From the **Members** page, navigate to a member in the table, select **...** and then select **View details.**
- 2. In the table, expand the respective row for organization, folder, or project where you want to view the member's assigned role and select **View** in the **Role** column.

View members associated with a folder or project

You can view members who have access to a specific folder or project.

Steps

- 1. Select **Administration > Identity and access.**
- 2. Select **Organization.**
- 3. From the **Organization** page, navigate to a project or folder in the table, select **...** and then select **Edit folder** or **Edit project.**
 - Select **Access** to view the members who have access to the folder or project.



Assign or modify member access

After a user signs up for NetApp Console, you can add them to your organization and assign them a role to provide access to resources. [Learn how to add members to your organization.](#)

You can adjust a member's access by adding or removing roles as needed.

Add an access role to a member

You typically assign a role when adding a member to your organization, but you can update it at any time by removing or adding roles.

You can assign a user an access role for your organization, folder, or project.

Members can have multiple roles within the same project and in different projects. For example, smaller organizations may assign all available access roles to the same user, while larger organizations may have users do more specialized tasks. Alternatively, you could also assign one user the Ransomware Resilience admin role at the organization level. In that example, the user would be able to perform Ransomware Resilience tasks on all projects within your organization.

Your access role strategy should align with the way you have organized your NetApp resources.

Steps

1. Select **Administration > Identity and access**.
2. Select **Members**.
3. Select one of the member tabs: **Users**, **Service accounts**, or **Federated groups**.
4. Select the actions menu **...** next to the member that you want to assign a role and select **Add a role**.
5. To add a role, complete the steps in the dialog box:

- **Select an organization, folder, or project:** Choose the level of your resource hierarchy that the member should have permissions for.

If you select the organization or a folder, the member will have permissions to everything that resides within the organization or folder.

- **Select a category:** Choose a role category. [Learn about access roles](#).
- **Select a Role:** Choose a role that provides the member with permissions for the resources that are associated with the organization, folder, or project that you selected.
- **Add role:** If you want to provide access to additional folders or projects within your organization, select **Add role**, specify another folder or project or role category, and then select a role category and a corresponding role.

6. Select **Add new roles**.

Change a member's assigned role


Change a member's roles to update their access.



Users must have at least one role assigned to them. You can't remove all roles from a user. If you need to remove all roles, you must delete the user from your organization.

Steps

1. Select **Administration > Identity and access**.
2. Select **Members**.
3. Select one of the member tabs: **Users**, **Service accounts**, or **Federated groups**.
4. From the **Members** page, navigate to a member in the table, select **...** and then select **View details**.

5. In the table, expand the respective row for organization, folder, or project where you want to change the member's assigned role and select **View** in the **Role** column to view the roles assigned to this member.
6. You can change an existing role for a member or remove a role.
 - a. To change a member's role, select **Change** next to the role you want to change. You can only change a role to a role within the same role category. For example, you can change from one data service role to another. Confirm the change.
 - b. To unassign a member's role, select  next to the role to remove the respective role from the member.. You'll be asked to confirm the removal.

Remove a member from your organization

Remove a member if they leave your organization.


When you remove a member, the system revokes their Console permissions but retains their Console and NetApp Support Site accounts.



Federated members

- Federated users automatically lose access to the NetApp Console when they are removed from your IdP. But you should still remove them from your Console organization to keep your member list up to date.
- If you remove a user from a federated group in your IdP, they lose the Console access associated with that group. However, they still retain any access associated with an explicit role assigned to them in the Console.

Steps

1. Select **Administration > Identity and access**.
2. Select **Members**.
3. Select one of the member tabs: **Users**, **Service accounts**, or **Federated groups**.
4. From the **Members** page, navigate to a member in the table, select  then select **Delete user**.
5. Confirm that you want to remove the member from your organization.

User security

Secure user access to your NetApp Console organization by managing member security settings. You can reset user passwords, manage multi-factor authentication (MFA), and recreate service account credentials.

Required access roles

Super admin, Org admin, or Folder or project admin (for folders and projects that they are administering).
 Link: reference-iam-predefined-roles.html[Learn about access roles].

Reset user passwords (local users only)

Org admins cannot reset user passwords for local users. However, they can instruct users to reset their own passwords.

Instruct a user to reset their password from the Console login page by selecting **Forgot password?**.



This option is not available for users in a federated organization.

Manage a user's multi-factor authentication (MFA)

If a user loses access to their MFA device, you can either remove or disable their MFA configuration.



Multi-factor authentication is only available for local users. Federated users cannot enable MFA.

Users must set up MFA again when they log in after removal. If the user temporarily loses access to their MFA device, they can use their saved recovery code to log in.

If they do not have their recovery code, temporarily disable MFA to allow login. When you disable MFA for a user, it is disabled for only eight hours and then re-enabled automatically. The user is allowed one login during that time without MFA. After the eight hours, the user must use MFA to log in.



To manage a user's multi-factor authentication, you must have an email address in the same domain as the affected user.

Steps

1. Select **Administration > Identity and access**.
2. Select **Members**.

The **Members** table lists the members of your organization.

3. From the **Members** page, navigate to a member in the table, select **...** and then select **Manage multi-factor authentication**.
4. Choose whether to remove or to disable the user's MFA configuration.

Recreate the credentials for a service account

You can create new credentials for a service if you lose or need to update them.

Creating new credentials deletes the old ones. You cannot use the old credentials.

Steps

1. Select **Administration > Identity and access**.
2. Select **Members**.
3. In the **Members** table, navigate to a service account, select **...** and then select **Recreate secrets**.
4. Select **Recreate**.
5. Download or copy the client ID and client secret.

The Console shows the client secret only once. Make sure you copy or download it and store it securely.

NetApp Console access roles

Learn about NetApp Console access roles

Identity and access management (IAM) in the NetApp Console provides predefined roles that you can assign to the members of your organization across different levels of your

resource hierarchy. Before you assign these roles, you should understand the permissions that each role includes. Roles fall into the following categories: platform, application, and data service.

Platform roles

Platform roles grant NetApp Console administration permissions, including role assignment and user management. The Console has several platform roles.

Platform role	Responsibilities
Organization admin	<p>Allows a user unrestricted access to all projects and folders within an organization, add members to any project or folder, as well as perform any task and use any data service that does not have an explicit role associated with it.</p> <p>Users with this role manage your organization by creating folders and projects, assigning roles, adding users, and managing systems if they have the proper credentials.</p> <p>This is the only access role that can create Console agents.</p>
Folder or project admin	<p>Allows a user unrestricted access to assigned projects and folders. Can add members to folders or projects they manage, as well as perform any task and use any data service or application on resources within the folder or project they are assigned.</p> <p>Folder or project admins cannot create Console agents.</p>
Federation admin	Allows a user to create and manage federations with the Console, which enables single-sign on (SSO).
Federation viewer	Allows a user to view existing federations with the Console. Cannot create or manage federations.
Partnership admin	Allows a user to create and manage partnerships.
Partnership viewer	Allows a user to view existing partnerships. Cannot create or manage partnerships.
Super admin	Gives the user a subset of admin roles. This role is designed for smaller organizations that may not need to distribute Console responsibilities across multiple users.
Super viewer	Gives the user a subset viewer roles. This role is designed for smaller organizations that may not need to distribute Console responsibilities across multiple users.

Application roles

The following is a list of roles in the application category. Each role grants specific permissions within its designated scope. Users without the required application or platform role cannot access the respective application.

Application role	Responsibilities
Google Cloud NetApp Volumes admin	Users with the Google Cloud NetApp Volumes role can discover and manage Google Cloud NetApp Volumes.
Google Cloud NetApp Volumes viewer	Users with the Google Cloud NetApp Volumes user role can view Google Cloud NetApp Volumes.
Keystone admin	Users with the Keystone admin role can create service requests. Allows users to monitor and view usage, resources, and admin details within the Keystone tenant they are accessing.
Keystone viewer	Users with the Keystone viewer role CANNOT create service requests. Allows users to monitor and view consumption, assets, and administrative information within the Keystone tenant they are accessing.
ONTAP Mediator setup role	Service accounts with the ONTAP Mediator setup role can create service requests. This role is required in a service account to configure an instance of the ONTAP Cloud Mediator .
Operation support analyst	Provides access to alerts and monitoring tools and ability to enter and manage support cases.
Storage admin	Administer storage health and governance functions, discover storage resources, as well as modify and delete existing systems.
Storage viewer	View storage health and governance functions, as well as view previously discovered storage resources. Cannot discover, modify, or delete existing storage systems.
System health specialist	Administer storage and health and governance functions, all permissions of the Storage admin except cannot modify or delete existing systems.

Data service roles

The following is a list of roles in the data service category. Each role grants specific permissions within its designated scope. Users who do not have the required data service role or a platform role will be unable to access the data service.

Data service role	Responsibilities
Backup and Recovery super admin	Perform any actions in NetApp Backup and Recovery.
Backup and Recovery admin	Perform backups to local snapshots, replicate to secondary storage, and back up to object storage.
Backup and Recovery restore admin	Restore workloads in the Backup and Recovery.
Backup and Recovery clone admin	Clone applications and data in the Backup and Recovery.
Backup and Recovery viewer	View Backup and Recovery information.
Disaster Recovery admin	Perform any actions in NetApp Disaster Recovery service.
Disaster Recovery failover admin	Perform failover and migrations.

Data service role	Responsibilities
Disaster Recovery application admin	Create replication plans, change replication plans, and start test failovers.
Disaster Recovery viewer	View information only.
Classification viewer	<p>Allows users to view NetApp Data Classification scan results.</p> <p>Users with this role can view compliance information and generate reports for resources that they have permission to access. These users can't enable or disable scanning of volumes, buckets, or database schemas. Classification does not have an admin role.</p>
Ransomware Resilience admin	Manage actions on the Protect, Alerts, Recover, Settings, and Reports tabs of NetApp Ransomware Resilience.
Ransomware Resilience viewer	View workload data, view alert data, download recovery data, and download reports in Ransomware Resilience.
Ransomware Resilience user behavior admin	Configure, manage, and view suspicious user behavior detection, alerts, and monitoring in Ransomware Resilience.
Ransomware Resilience user behavior viewer	View suspicious user behavior alerts and insights in Ransomware Resilience.
SnapCenter admin	<p>Provides the ability to back up snapshots from on-premises ONTAP clusters using NetApp Backup and Recovery for applications. A member who has this role can complete the following actions:</p> <ul style="list-style-type: none"> * Complete any action from Backup and Recovery > Applications * Manage all systems in the projects and folders for which they have permissions * Use all NetApp Console services <p>SnapCenter does not have a viewer role.</p>

Related links

- [Learn about NetApp Console identity and access management](#)
- [Get started with NetApp Console IAM](#)
- [Manage NetApp Console members and their permissions](#)
- [Learn about the API for NetApp Console IAM](#)

NetApp Console platform access roles

Assign platform roles to users to grant permissions to manage the NetApp Console, assign roles, add users, create Console agents, and manage federations.

Example for organization roles for a large multi-national organization

XYZ Corporation organizes data storage access by region—North America, Europe, and Asia-Pacific—providing regional control with centralized oversight.

The **Organization admin** in XYZ Corporation's Console creates an initial organization and separate folders for

each region. The **Folder or project admin** for each region organizes projects (with associated resources) within the region's folder.

Regional admins with the **Folder or project admin** role actively manage their folders by adding resources and users. These regional admins can also add, remove, or rename folders and projects they manage. The **Organization admin** inherits permissions for any new resources, maintaining visibility of storage usage across the entire organization.

Within the same organization, one user is assigned the **Federation admin** role to manage the federation of the organization with their corporate IdP. This user can add or remove federated organizations, but cannot manage users or resources within the organization. The **Organization admin** assigns a user the **Federation viewer** role to check federation status and view federated organizations.

The following tables indicate the actions that each Console platform role can perform.

Organization administration roles

Task	Organization admin	Folder or project admin
Create agents	Yes	No
Create, modify or delete systems from the Console (add or discover systems)	Yes	Yes
Create folders and projects, including deleting	Yes	No
Rename existing folders and projects	Yes	Yes
Assign roles and add users	Yes	Yes
Associate resources with folders and projects	Yes	Yes
Associate agents with folders and projects	Yes	No
Remove agents from folders and projects	Yes	No
Manage agents (edit certificates, settings, and so on)	Yes	No
Manage credentials from Administration > Credentials	Yes	Yes
Create, manage, and view federations	Yes	No
Register for support and submit cases through the Console	Yes	Yes
Use data services that are not associated with an explicit access role	Yes	Yes
View the Audit page and notifications	Yes	Yes

Federation roles

Task	Federation admin	Federation viewer
Create a federation	Yes	No
Verify a domain	Yes	No
Add a domain to a federation	Yes	No
Disable and delete federations	Yes	No

Task	Federation admin	Federation viewer
Test federations	Yes	No
View federations and their details	Yes	Yes

Partnership roles

Task	Partnership admin	Partnership viewer
Can create a partnership	Yes	No
Assign roles to partner members	Yes	No
Can add members to a partnership	Yes	No
Can view organization partnership details	Yes	Yes

Super admin and viewer roles

The **Super admin** role provides full access to manage Console features, storage, and data services. This role suits those overseeing administration and governance. In contrast, the **Super viewer** role offers read-only access, ideal for auditors or stakeholders who need visibility without making changes.

Organizations should use **Super admin** access sparingly to minimize security risks and align with the principle of least privilege. Most organizations should assign fine-grained roles with only the necessary permissions to reduce risk and improve auditability.

Example for super roles

ABC Corporation has a small team of five that leverages the NetApp Console for data services and storage management. Instead of distributing multiple roles, they assign the **Super admin** role to two senior team members who handle all administrative tasks, including user management and resource configuration. The remaining three team members are assigned the **Super viewer** role, allowing them to monitor storage health and data service status without the ability to modify settings.

Role	Inherited roles
Super admin	<ul style="list-style-type: none"> • Organization admin • Folder or project admin • Federation admin • Partnership admin • Ransomware Resilience admin • Disaster recovery admin • Backup super admin • Storage admin • Keystone admin • Google Cloud NetApp Volumes admin

Role	Inherited roles
Super viewer	<ul style="list-style-type: none"> • Organization viewer • Federation viewer • Partnership viewer • Ransomware Resilience viewer • Disaster recovery viewer • Backup viewer • Storage viewer • Keystone viewer • Google Cloud NetApp Volumes viewer

Application roles

Google Cloud NetApp Volumes roles in NetApp Console

You can assign the following role to users to provide them access to the Google Cloud NetApp Volumes in the NetApp Console.

Google Cloud NetApp Volumes uses the following role:

- **Google Cloud NetApp Volumes admin:** Discover and manage Google Cloud NetApp Volumes in the Console.
- **Google Cloud NetApp Volumes viewer:** View Google Cloud NetApp Volumes in the Console.

Keystone access roles in NetApp Console

Keystone roles provide access to the Keystone dashboards and allow users to view and manage their Keystone subscription. There are two Keystone roles: Keystone admin and Keystone viewer. The main difference between the two roles is the actions they can take in Keystone. The Keystone admin role is the only role that is allowed to create service requests or modify subscriptions.

Example for Keystone roles in NetApp Console

XYZ Corporation has four storage engineers from different departments who view Keystone subscription information. Although all of these users need to monitor the Keystone subscription, only the team lead is allowed to make service requests. Three of the team members are given the **Keystone viewer** role, while the team lead is given the **Keystone admin** role so that there is a point of control over service requests for the company.

The following table indicates the actions that each Keystone role can perform.

Feature and action	Keystone admin	Keystone viewer
View the following tabs: Subscription, Assets, Monitor, and Administration	Yes	Yes

Feature and action	Keystone admin	Keystone viewer
Keystone subscription page:		
View subscriptions	Yes	Yes
Amend or renew subscriptions	Yes	No
Keystone assets page:		
View assets	Yes	Yes
Manage assets	Yes	No
Keystone alerts page:		
View alerts	Yes	Yes
Manage alerts	Yes	No
Create alerts for self	Yes	Yes
Licenses and subscriptions:		
Can view licenses and subscriptions	Yes	Yes
Keystone reports page:		
Download reports	Yes	Yes
Manage reports	Yes	Yes
Create reports for self	Yes	Yes
Service requests:		
Create service requests	Yes	No
View service requests created by any user within the Organization	Yes	Yes

Operational support analyst access role for NetApp Console

You can assign the Operational support analyst role to users to provide them access to alerts and monitoring. Users with this role can also open support cases.

Operational support analyst

Task	Can perform
Manage own user credentials from Settings > Credentials	Yes
View discovered resources	Yes
Register for support and submit cases through the Console	Yes
View the Audit page and notifications	Yes
View, download, and configure alerts	Yes

Storage access roles for NetApp Console

You can assign the following roles to users to provide them access to the storage management features in the NetApp Console. You can assign users an administrative role to manage storage or a viewer role for monitoring.



These roles are not available from the NetApp Console partnership API.

Administrators can assign storage roles to users for the following storage resources and features:

Storage resources:

- On-premises ONTAP clusters
- StorageGRID
- E-Series

Console services and features:

- Digital advisor
- Software updates
- Lifecycle planning
- Sustainability

Example for storage roles in NetApp Console

XYZ Corporation, a multinational company, has a large team of storage engineers and storage administrators. They allow this team to manage storage assets for their regions while limiting access to core Console tasks like user management, agent creation, and license management.

Within a team of 12, two users are given the **Storage viewer** role which allows them to monitor the storage resources associated with the Console projects they are assigned to. The remaining nine are given the **Storage admin** role which includes the ability to manage software updates, access ONTAP System Manager through the Console, as well as discover storage resources (add systems). One person on the team is given the **System health specialist** role so they can manage the health of the storage resources in their region, but not modify or delete any systems. This person can also perform software updates on the storage resources for projects they are assigned.

The organization has two additional users with the **Organization admin** role who can manage all aspects of

the Console, including user management, agent creation, and license management, as well as several users with the **Folder or project admin** role who can perform Console administration tasks for the folders and projects they are assigned to.

The following table shows the actions each storage role performs.

Feature and action	Storage admin	System health specialist	Storage viewer
Storage Management:			
Discover new resources (create systems)	Yes	Yes	No
View discovered systems	Yes	Yes	No
Delete systems from the Console	Yes	No	No
Modify systems	Yes	No	No
Create agents	No	No	No
Digital advisor			
View all pages and functions	Yes	Yes	Yes
Licenses and subscriptions			
View all pages and functions	No	No	No
Software updates			
View landing page and recommendations	Yes	Yes	Yes
Review potential version recommendations and key benefits	Yes	Yes	Yes
View update details for a cluster	Yes	Yes	Yes
Run pre-update checks and download upgrade plan	Yes	Yes	Yes
Install software updates	Yes	Yes	No
Lifecycle planning			
Review capacity planning status	Yes	Yes	Yes
Choose next action (best practice, tier)	Yes	No	No
Tier cold data to cloud storage and free up storage	Yes	Yes	No

Feature and action	Storage admin	System health specialist	Storage viewer
Set up reminders	Yes	Yes	Yes
Sustainability			
View dashboard and recommendations	Yes	Yes	Yes
Download report data	Yes	Yes	Yes
Edit carbon mitigation percentage	Yes	Yes	No
Fix recommendations	Yes	Yes	No
Defer recommendations	Yes	Yes	No
System manager access			
May enter credentials	Yes	Yes	No
Credentials			
User credentials	Yes	Yes	No

Data services roles

NetApp Backup and Recovery roles in NetApp Console

You can assign the following roles to users to provide them access to NetApp Backup and Recovery within the Console. Backup and Recovery roles give you the flexibility to assign users a role specific to the tasks they need to accomplish within your organization. How you assign roles depends on your own business and storage management practices.

The service uses the following roles that are specific to NetApp Backup and Recovery.

- **Backup and Recovery super admin:** Perform any actions in NetApp Backup and Recovery.
- **Backup and Recovery Backup admin:** Perform backups to local snapshots, replicate to secondary storage, and back up to object storage actions in NetApp Backup and Recovery.
- **Backup and Recovery Restore admin:** Restore workloads using NetApp Backup and Recovery.
- **Backup and Recovery Clone admin:** Clone applications and data using NetApp Backup and Recovery.
- **Backup and Recovery viewer:** View information in NetApp Backup and Recovery, but not perform any actions.

For details about all NetApp Console access roles, see [the Console setup and administration documentation](#).

Roles used for common actions

The following table indicates the actions that each NetApp Backup and Recovery role can perform for all workloads.

Feature and action	Backup and Recovery super admin	Backup and Recovery backup admin	Backup and Recovery restore admin	Backup and Recovery clone admin	Backup and Recovery viewer
Add, edit, or delete hosts	Yes	No	No	No	No
Install plugins	Yes	No	No	No	No
Add credentials (host, instance, vCenter)	Yes	No	No	No	No
View dashboard and all tabs	Yes	Yes	Yes	Yes	Yes
Start free trial	Yes	No	No	No	No
Initiate discovery of workloads	No	Yes	Yes	Yes	No
View license information	Yes	Yes	Yes	Yes	Yes
Activate license	Yes	No	No	No	No
View hosts	Yes	Yes	Yes	Yes	Yes
Schedules:					
Activate schedules	Yes	Yes	Yes	Yes	No
Suspend schedules	Yes	Yes	Yes	Yes	No
Policies and protection:					
View protection plans	Yes	Yes	Yes	Yes	Yes
Create, modify, or delete protection plans	Yes	Yes	No	No	No
Restore workloads	Yes	No	Yes	No	No
Create, split, or delete clones	Yes	No	No	Yes	No
Create, modify, or delete policy	Yes	Yes	No	No	No

Feature and action	Backup and Recovery super admin	Backup and Recovery backup admin	Backup and Recovery restore admin	Backup and Recovery clone admin	Backup and Recovery viewer
Reports:					
View reports	Yes	Yes	Yes	Yes	Yes
Create reports	Yes	Yes	Yes	Yes	No
Delete reports	Yes	No	No	No	No
Import from SnapCenter and manage host:					
View imported SnapCenter data	Yes	Yes	Yes	Yes	Yes
Import data from SnapCenter	Yes	Yes	No	No	No
Manage (migrate) host	Yes	Yes	No	No	No
Configure settings:					
Configure log directory	Yes	Yes	Yes	No	No
Associate or remove instance credentials	Yes	Yes	Yes	No	No
Buckets:					
View buckets	Yes	Yes	Yes	Yes	Yes
Create, edit, or delete bucket	Yes	Yes	No	No	No

Roles used for workload-specific actions

The following table indicates the actions that each NetApp Backup and Recovery role can perform for specific workloads.

Kubernetes workloads

This table indicates the actions that each NetApp Backup and Recovery role can perform for actions specific to Kubernetes workloads.

Feature and action	Backup and Recovery super admin	Backup and Recovery backup admin	Backup and Recovery restore admin	Backup and Recovery viewer
View clusters, namespaces, storage classes, and API resources	Yes	Yes	Yes	Yes

Feature and action	Backup and Recovery super admin	Backup and Recovery backup admin	Backup and Recovery restore admin	Backup and Recovery viewer admin
Add new Kubernetes clusters	Yes	Yes	No	No
Update cluster configurations	Yes	No	No	No
Remove clusters from management	Yes	No	No	No
View applications	Yes	Yes	Yes	Yes
Create and define new applications	Yes	Yes	No	No
Update application configurations	Yes	Yes	No	No
Remove applications from management	Yes	Yes	No	No
View protected resources and backup status	Yes	Yes	Yes	Yes
Create backups and protect applications with policies	Yes	Yes	No	No
Unprotect apps and delete backups	Yes	Yes	No	No
View recovery points and resource viewer results	Yes	Yes	Yes	Yes
Restore applications from recovery points	Yes	No	Yes	No
View Kubernetes backup policies	Yes	Yes	Yes	Yes
Create Kubernetes backup policies	Yes	Yes	Yes	No
Update backup policies	Yes	Yes	Yes	No
Delete backup policies	Yes	Yes	Yes	No
View execution hooks and hook sources	Yes	Yes	Yes	Yes
Create execution hooks and hook sources	Yes	Yes	Yes	No

Feature and action	Backup and Recovery super admin	Backup and Recovery backup admin	Backup and Recovery restore admin	Backup and Recovery viewer admin
Update execution hooks and hook sources	Yes	Yes	Yes	No
Delete execution hooks and hook sources	Yes	Yes	Yes	No
View execution hook templates	Yes	Yes	Yes	Yes
Create execution hook templates	Yes	Yes	Yes	No
Update execution hook templates	Yes	Yes	Yes	No
Delete execution hook templates	Yes	Yes	Yes	No
View workload summary and analytics dashboards	Yes	Yes	Yes	Yes
View StorageGRID buckets and storage targets	Yes	Yes	Yes	Yes

NetApp Disaster Recovery roles in NetApp Console

You can assign the following roles to users to provide them access to NetApp Disaster Recovery within the Console. Disaster Recovery roles give you the flexibility to assign users a role specific to the tasks they need to accomplish within your organization. How you assign roles depends on your own business and storage management practices.

Disaster Recovery uses the following roles:

- **Disaster recovery admin:** Perform any actions.
- **Disaster recovery failover admin:** Perform failover and migrations.
- **Disaster recovery application admin:** Create replication plans. Modify replication plans. Start test failovers.
- **Disaster recovery viewer:** View information only.

The following table indicates the actions that each role can perform.

Feature and action	Disaster recovery admin	Disaster recovery failover admin	Disaster recovery application admin	Disaster recovery viewer
View dashboard and all tabs	Yes	Yes	Yes	Yes
Start free trial	Yes	No	No	No

Feature and action	Disaster recovery admin	Disaster recovery failover admin	Disaster recovery application admin	Disaster recovery viewer
Initiate discovery of workloads	Yes	No	No	No
View license information	Yes	Yes	Yes	Yes
Activate license	Yes	No	Yes	No
On the Sites tab:				
View sites	Yes	Yes	Yes	Yes
Add, modify, or delete sites	Yes	No	No	No
On the Replication plans tab:				
View replication plans	Yes	Yes	Yes	Yes
View replication plan details	Yes	Yes	Yes	Yes
Create or modify replication plans	Yes	Yes	Yes	No
Create reports	Yes	No	No	No
View snapshots	Yes	Yes	Yes	Yes
Perform failover tests	Yes	Yes	Yes	No
Perform failovers	Yes	Yes	No	No
Perform failbacks	Yes	Yes	No	No
Perform migrations	Yes	Yes	No	No
On the Resource groups tab:				
View resource groups	Yes	Yes	Yes	Yes
Create, modify, or delete resource groups	Yes	No	Yes	No
On the Job Monitoring tab:				
View jobs	Yes	No	Yes	Yes
Cancel jobs	Yes	Yes	Yes	No

Ransomware Resilience roles provide users access to NetApp Ransomware Resilience. Ransomware Resilience supports the following roles:

Baseline roles

- Ransomware Resilience admin - Configure Ransomware Resilience settings; investigate and respond to encryption alerts
- Ransomware Resilience viewer - View encryption incidents, reports, and discovery settings

User behavior activity roles

[Suspicious user activity detection](#) alerts provide visibility into data such as file activity events; these alerts include file names and file actions (such as Read, Write, Delete, Rename) performed by the user. To limit the visibility of this data, only users with these roles can manage or view these alerts.

- Ransomware Resilience user behavior admin - Activate suspicious user activity detection, investigate and respond to suspicious user activity alerts
- Ransomware Resilience user behavior viewer - View suspicious user activity alerts



User behavior roles are not standalone roles; they are designed to be added to Ransomware Resilience admin or viewer roles. For more information, see [User behavior roles](#).

Consult the following tables for detailed descriptions of each role.

Baseline roles

The following table describes the actions available to the Ransomware Resilience admin and viewer roles.

Feature and action	Ransomware Resilience admin	Ransomware Resilience viewer
View dashboard and all tabs	Yes	Yes
On dashboard, update recommendation status	Yes	No
Start free trial	Yes	No
Initiate discovery of workloads	Yes	No
Initiate rediscovery of workloads	Yes	No
On the Protect tab:		
Add, modify, or delete protection plans for <i>encryption</i> policies	Yes	No
Protect workloads	Yes	No
Identify exposure to sensitive data with Data Classification	Yes	No

Feature and action	Ransomware Resilience admin	Ransomware Resilience viewer
List protection plans and details	Yes	Yes
List protection groups	Yes	Yes
View protection group details	Yes	Yes
Create, edit, or delete protection groups	Yes	No
Download data	Yes	Yes
On the Alerts tab:		
View encryption alerts and alert details	Yes	Yes
Edit encryption incident status	Yes	No
Mark encryption alert for recovery	Yes	No
View encryption incident details	Yes	Yes
Dismiss or resolve encryption incidents	Yes	No
Get full list of impacted files in encryption event	Yes	No
Download encryption event alerts data	Yes	Yes
Block user (with Workload Security agent configuration)	Yes	No
On the Recover tab:		
Download impacted files from encryption event	Yes	No
Restore workload from encryption event	Yes	No
Download recovery data from encryption event	Yes	Yes
Download reports from encryption event	Yes	Yes
On the Settings tab:		
Add or modify backup destinations	Yes	No
List backup destinations	Yes	Yes

Feature and action	Ransomware Resilience admin	Ransomware Resilience viewer
View connected SIEM targets	Yes	Yes
Add or modify SIEM targets	Yes	No
Configure readiness drill	Yes	No
Start, reset, or edit readiness drill	Yes	No
Review readiness drill status	Yes	Yes
Update discovery configuration	Yes	No
View discovery configuration	Yes	Yes
On the Reports tab:		
Download reports	Yes	Yes

User behavior roles

To configure suspicious user behavior settings and respond to alerts, a user must have the Ransomware Resilience user behavior admin role. To only view suspicious user behavior alerts, a user should have the Ransomware Resilience user behavior viewer role.

User behavior roles should be conferred on users with existing Ransomware Resilience admin or viewer privileges who need access to [suspicious user activity settings and alerts](#). A user with the Ransomware Resilience admin role, for example, should receive the Ransomware Resilience user behavior admin role to configure user activity agents and block or unblock users. The Ransomware Resilience user behavior admin role should not be conferred on a Ransomware Resilience viewer.



To activate suspicious user activity detection, you must have the Console Organization admin role.

The following table describes the actions available to the Ransomware Resilience user behavior admin and viewer roles.

Feature and action	Ransomware Resilience user behavior admin	Ransomware Resilience user behavior viewer
On the Settings tab:		
Create, modify, or delete user activity agent	Yes	No
Create or delete user directory connector	Yes	No
Pause or resume data collector	Yes	No

Feature and action	Ransomware Resilience user behavior admin	Ransomware Resilience user behavior viewer
Run data breach readiness drill	Yes	No
On the Protect tab:		
Add, modify, or delete protection plans for <i>suspicious user behavior</i> policies	Yes	No
On the Alerts tab:		
View user activity alerts and alert details	Yes	Yes
Edit user activity incident status	Yes	No
Mark user activity alert for recovery	Yes	No
View user activity incident details	Yes	Yes
Dismiss or resolve user activity incidents	Yes	No
Get full list of impacted files by suspicious user	Yes	Yes
Download user activity event alerts data	Yes	Yes
Block or unblock user	Yes	No
On the Recover tab:		
Download impacted files for user activity event	Yes	No
Restore workload from user activity event	Yes	No
Download recovery data from user activity event	Yes	Yes
Download reports from user activity event	Yes	Yes

Identity and access API

Organization and project IDs

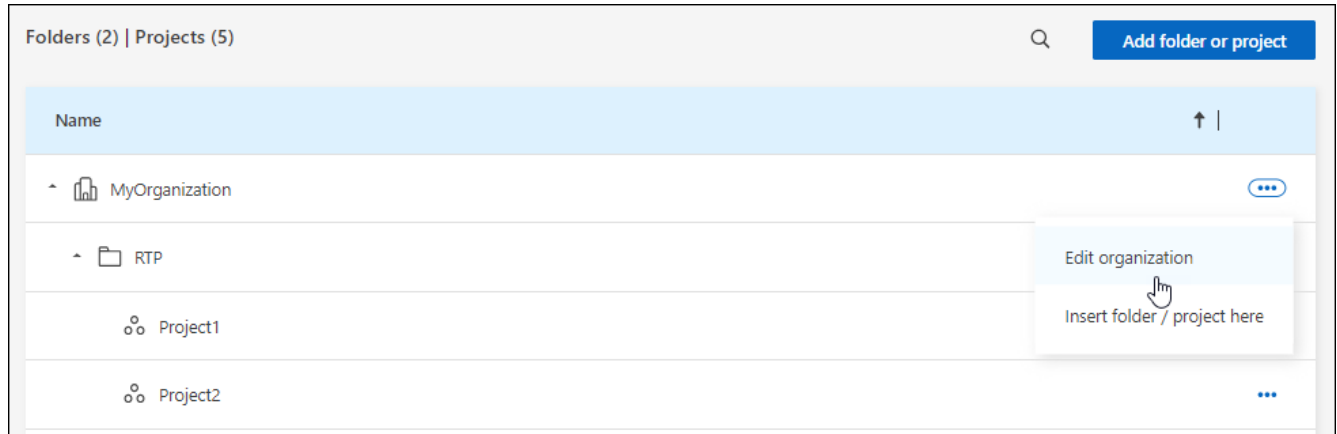
Your NetApp Console organization has a name and an ID. You can choose a name for your organization to help identify it. You may also need to retrieve the organization ID for certain integrations.

Rename your organization

You can rename your organization. This is helpful if you support more than organization.

Steps

1. Select **Administration > Identity and access**.
2. Select **Organization**.
3. From the **Organization** page, navigate to the first row in the table, select **...** and then select **Edit organization**.



4. Enter a new organization name and select **Apply**.

Get the organization ID

The organization ID is used for certain integrations with the Console.

You can view the organization ID from the Organizations page and copy it to the clipboard for your needs.

Steps

1. Select **Administration > Identity and access > Organization**.
2. On the **Organization** page, look for your organization ID in the summary bar and copy it to the clipboard. You can save this for use later or copy it directly to where you need to use it.

Obtain the ID for a project

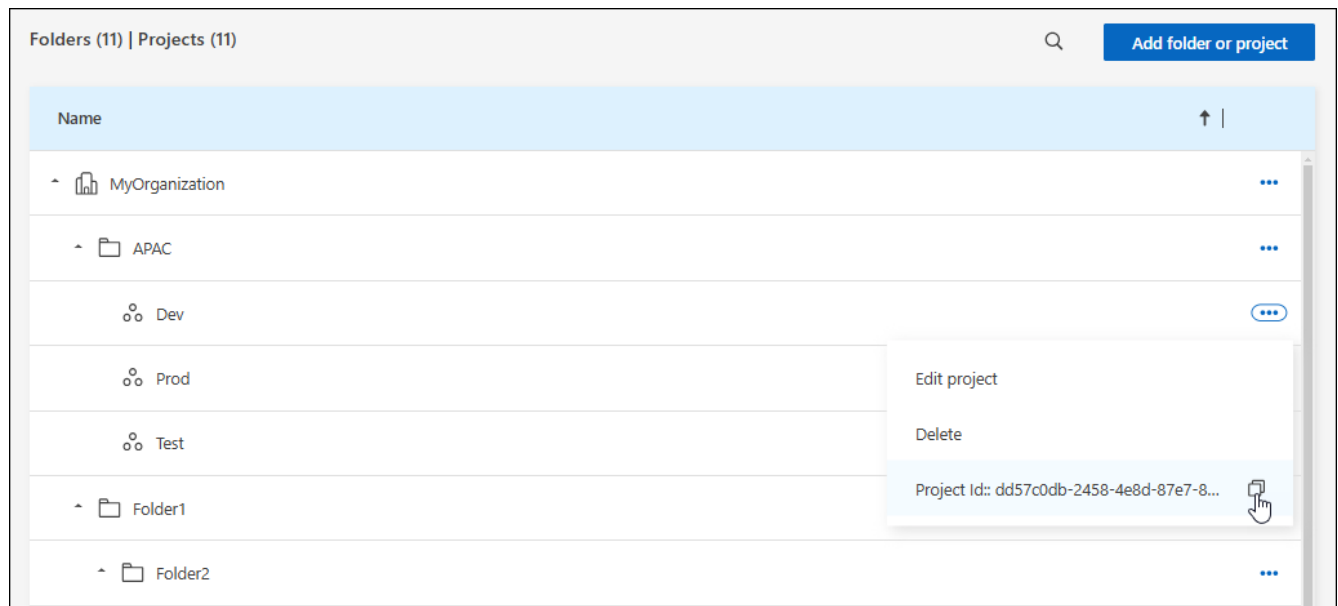
You'll need to obtain the ID for a project if you are using the API. For example, when creating a Cloud Volumes ONTAP system.

Steps

1. From the **Organization** page, navigate to a project in the table and select **...**

The project ID displays.

2. To copy the ID, select the copy button.



Related information

- [Learn about identity and access management](#)
- [Get started with identity and access](#)
- [Learn about the API for identity and access](#)

Security and compliance

Identity federation

Enable single sign-on by using identity federation with NetApp Console

Single-sign on (federation) simplifies the login process and enhances security by allowing users to log in to the NetApp Console using their corporate credentials. You can enable single sign-on (SSO) with your identity provider (IdP) or with the NetApp Support site.

Required role

Organization admin, Federation admin, Federation viewer. [Learn more about access roles.](#)

Identity federation with NetApp Support Site

Federating with the NetApp Support Site allows users to log in to the Console, Active IQ Digital Advisor, and other associated apps using the same credentials.



If you federate with the NetApp Support Site, you can't also federate with your corporate identity management provider. Choose which one works best for your organization.

Steps

1. Download and complete the [NetApp Federation Request Form](#).
2. Submit the form to the email address specified in the form.

The NetApp support team reviews and processes your request.

Set up a federated connection with your identity provider

You can set up a federated connection with your identity provider to enable single sign-on (SSO) for the Console. The process involves configuring your identity provider to trust NetApp as a service provider and then creating the connection in the Console.



If you previously configured federation using NetApp Cloud Central (an external application to the Console), you need to import your federation using the Federation page to manage it within the Console. [Learn how to import your federation.](#)

Supported identity providers

NetApp supports the following protocols and identity providers for federation:

Protocols

- Security Assertion Markup Language (SAML) identity providers
- Active Directory Federation Services (AD FS)

Identity providers

- Microsoft Entra ID
- PingFederate

Federation with NetApp Console workflow

NetApp supports service provider-initiated (SP-initiated) SSO only. You need to first configure the identity provider to trust NetApp as a service provider. Then, you can create a connection in the Console that uses the identity provider's configuration.

You can federate with your email domain or with a different domain that you own. To federate with a domain different from your email domain, first verify you own the domain.

1

Verify your domain (if not using your email domain)

To federate with a domain different from your email domain, verify that you own it. You can federate your email domain without any extra steps.

2

Configure your IdP to trust NetApp as a service provider

Configure your identity provider to trust NetApp by creating a new application and providing details like the ACS URL, Entity ID or other credential information. Service provider information varies by identity provider, so refer to the documentation for your specific identity provider for details. You'll need to work with your IdP administrator to complete this step.

3

Create the federated connection in the Console

Provide the SAML metadata URL or file from your identity provider to create the connection. This information is used to establish the trust relationship between the Console and your identity provider. The information you provide depends on the IdP that you are using. For example, if you're using Microsoft Entra ID, you need to provide the client ID, secret, and domain.

4

Test your federation in the Console

Test your federated connection before enabling it. Use the test option on the Federation page in the Console to verify that your test user can authenticate successfully. If the test is successful, you can enable the connection.

5

Enable your connection in the Console

After you enable the connection, users can log in to the Console using their corporate credentials.

Review the topic for your respective protocol or IdP to get started:

- [Set up a federated connection with AD FS](#)
- [Set up a federated connection with Microsoft Entra ID](#)
- [Set up a federated connection with PingFederate](#)
- [Set up a federated connection with a SAML identity provider](#)

Domain verification

Verify the email domain for your federated connection

If you want to federate with a domain that is different than your email domain, you must first verify that you own the domain. You can only use verified domains for federation.

Required roles

The Federation admin role is required to make create and manage federations. Federation viewer can view the Federation page. [Learn more about access roles.](#)

Verifying your domain involves adding a TXT record to your domain's DNS settings. This record is used to prove that you own the domain and allows the NetApp Console to trust the domain for federation. You may need to coordinate with your IT or network administrator to complete this step.

Steps

1. Select **Administration > Identity and access**.
2. Select **Federation** to view the **Federations** page.
3. Select **Configure new federation**.
4. Select **Verify domain ownership**.
5. Enter the domain that you want to verify and select **Continue**.
6. Copy the TXT record that is provided.
7. Go to your domain's DNS settings and configure the TXT value that was provided as a TXT record for your domain. Work with your IT or network administrator if needed.
8. After the TXT record is added, return to the Console and select **Verify**.

Configure federations

Federate NetApp Console with Active Directory Federation Services (AD FS)

Federate your Active Directory Federation Services (AD FS) with the NetApp Console to

enable single sign-on (SSO) for the NetApp Console. This allows users to log in to the Console using their corporate credentials.

Required roles

The Federation admin role is required to make create and manage federations. Federation viewer can view the Federation page. [Learn more about access roles.](#)



You can federate with your corporate IdP or with the NetApp Support Site. NetApp recommends choosing one or the other, but not both.

NetApp supports service provider-initiated (SP-initiated) SSO only. First, configure the identity provider to trust the NetApp Console as a service provider. Then, create a connection in the Console using your identity provider's configuration.

You can set up federation with your AD FS server to enable single sign-on (SSO) for NetApp Console. The process involves configuring your AD FS to trust the Console as a service provider and then creating the connection in the NetApp Console.

Steps

1. Select **Administration > Identity and access**.
2. Select **Federation** to view the **Federations** page.
3. Select **Configure new federation**.
4. Enter your domain details:
 - a. Choose whether you want to use a verified domain or your email domain. The email domain is the domain associated with the account you are logged in with.
 - b. Enter the name of the federation you are configuring.
 - c. If you choose a verified domain, select the domain from the list.
5. Select **Next**.
6. For your connection method, choose **Protocol** and then select **Active Directory Federation Services (AD FS)**.
7. Select **Next**.
8. Create a Relying Party Trust in your AD FS server. You can use PowerShell or manually configure it on your AD FS server. Consult the AD FS documentation for details on how to create a relying party trust.
 - a. Create the trust using PowerShell by using following script:

```
(new-object Net.WebClient -property @{Encoding = [Text.Encoding]
::UTF8}) .DownloadString("https://raw.githubusercontent.com/auth0/AD FS-
auth0/master/AD FS.ps1") | iex
AddRelyingParty "urn:auth0:netapp-cloud-account" "https://netapp-
cloud-account.auth0.com/login/callback"
```

- b. Alternatively, you can create the trust manually in the AD FS management console. Use the following NetApp Console values when creating the trust:
 - When creating the Relying Trust Identifier, use the **YOUR_TENANT** value: netapp-cloud-account

- When you select **Enable support for the WS-Federation**, use the **YOUR_AUTH0_DOMAIN** value: `netapp-cloud-account.auth0.com`

c. After creating the trust, copy the metadata URL from your AD FS server or download the federation metadata file. You'll need this URL or file to complete the connection in the Console.

NetApp recommends using the metadata URL to let the NetApp Console automatically retrieve the latest AD FS configuration. If you download the federation metadata file, you will need to update it manually in the NetApp Console whenever there are changes to your AD FS configuration.

9. Return to the Console, and select **Next** to create the connection.
10. Create the connection with AD FS.
 - a. Enter the **AD FS URL** that you copied from your AD FS server in the previous step or upload the federation metadata file that you downloaded from your AD FS server.
11. Select **Create connection**. Creating the connection might take a few seconds.
12. Select **Next**.
13. Select **Test connection** to test your connection. You are directed to a login page for your IdP server. Log in with your IdP credentials. After you log in, go back to the Console to enable the connection.



When using the Console in restricted mode, copy the URL to either an incognito browser window or a separate browser to log in to your IdP.

14. In the Console, select **Next** to review the summary page.
15. Set up notifications.

Choose between seven days or 30 days. The system emails expiry notifications and shows them in the Console to any user with the following roles: Super admin, Org admin, Federation admin, and Federation viewer.

16. Review the federation details and then select **Enable federation**.
17. Select **Finish** to complete the process.

After you enable the federation, users log in to the NetApp Console using their corporate credentials.

Federate NetApp Console with Microsoft Entra ID

Federate with your Microsoft Entra ID IdP provider to enable single sign-on (SSO) for the NetApp Console. This allows users to log in using their corporate credentials.

Required roles

The Federation admin role is required to make create and manage federations. Federation viewer can view the Federation page. [Learn more about access roles.](#)



You can federate with your corporate IdP or with the NetApp Support Site. NetApp recommends choosing one or the other, but not both.

NetApp supports service provider-initiated (SP-initiated) SSO only. You need to first configure the identity provider to trust NetApp as a service provider. Then, you can create a connection in the Console that uses the identity provider's configuration.

You can set up a federated connection with Microsoft Entra ID to enable single sign-on (SSO) for the Console .

The process involves configuring your Microsoft Entra ID to trust the Console as a service provider and then creating the connection in the Console.

Steps

1. Select **Administration > Identity and access**.
2. Select **Federation** to view the **Federations** page.
3. Select **Configure new federation**.

Domain details

4. Enter your domain details:
 - a. Choose whether you want to use a verified domain or your email domain. The email domain is the domain associated with the account you are logged in with.
 - b. Enter the name of the federation you are configuring.
 - c. If you choose a verified domain, select the domain from the list.
5. Select **Next**.

Connection method

6. For your connection method, choose **Provider** and then select **Microsoft Entra ID**.
7. Select **Next**.

Configuration instructions

1. Configure your Microsoft Entra ID to trust NetApp as a service provider. You need to do this step on your Microsoft Entra ID server.
 - a. Use the following values when registering your Microsoft Entra ID app to trust the Console:
 - For the **Redirect URL** , use <https://services.cloud.netapp.com>
 - For the **Reply URL**, use <https://netapp-cloud-account.auth0.com/login/callback>
 - b. Create a client secret for your Microsoft Entra ID app. You'll need to provide the client ID, the client secret, and the Entra ID domain name to complete the federation.
2. Return to the Console, and select **Next** to create the connection.

Create connection

1. Create the connection with Microsoft Entra ID
 - a. Enter the client ID and Client secret that you created in the previous step.
 - b. Enter the Microsoft Entra ID domain name.
2. Select **Create connection**. The system creates the connection in a few seconds.

Test and enable the connection

1. Select **Next**.
2. Select **Test connection** to test your connection. You are directed to a login page for your IdP server. Log in with your IdP credentials. After you log in, go back to the Console to enable the connection.



When using the Console in restricted mode, copy the URL to either an incognito browser window or a separate browser to log in to your IdP.

3. In the Console, select **Next** to review the summary page.
4. Set up notifications.

Choose between seven days or 30 days. The system emails expiry notifications and shows them in the Console to any user with the following roles: Super admin, Org admin, Federation admin, and Federation viewer.

5. Review the federation details and then select **Enable federation**.
6. Select **Finish** to complete the process.

After you enable the federation, users log in to the NetApp Console using their corporate credentials.

Federate NetApp Console with PingFederate

Federate with your PingFederate IdP provider to enable single sign-on (SSO) for the NetApp Console. This allows users to log in using their corporate credentials.

Required roles

The Federation admin role is required to make create and manage federations. Federation viewer can view the Federation page. [Learn more about access roles](#).



You can federate with your corporate IdP or with the NetApp Support Site. NetApp recommends choosing one or the other, but not both.

NetApp supports service provider-initiated (SP-initiated) SSO only. You need to first configure the identity provider to trust NetApp as a service provider. Then, you can create a connection in the Console that uses the identity provider's configuration.

You can set up a federated connection with PingFederate to enable single sign-on (SSO) for the Console . The process involves configuring your PingFederate server to trust the Console as a service provider and then creating the connection in the Console .

Steps

1. Select **Administration > Identity and access**.
2. Select **Federation** to view the **Federations** page.
3. Select **Configure new federation**.
4. Enter your domain details:
 - a. Choose whether you want to use a verified domain or your email domain. The email domain is the domain associated with the account you are logged in with.
 - b. Enter the name of the federation you are configuring.
 - c. If you choose a verified domain, select the domain from the list.
5. Select **Next**.
6. For your connection method, choose **Provider** and then select **PingFederate**.
7. Select **Next**.

8. Configure your PingFederate server to trust NetApp as a service provider. You need to do this step on your PingFederate server.
 - a. Use the following values when configuring PingFederate to trust the NetApp Console:
 - For the **Reply URL** or **Assertion Consumer Service (ACS) URL**, use <https://netapp-cloud-account.auth0.com/login/callback>
 - For the **Logout URL**, use <https://netapp-cloud-account.auth0.com/logout>
 - For **Audience/Entity ID**, use `urn:auth0:netapp-cloud-account:<fed-domain-name-saml>` where `<fed-domain-name-pingfederate>` is the domain name for the federation. For example, if your domain is `example.com`, the Audience/Entity ID would be `urn:auth0:netappcloud-account:fed-example-com-pingfederate`.
 - b. Copy the PingFederate server URL. You will need this URL when creating the connection in the Console.
 - c. Download the X.509 certificate from your PingFederate server. It needs to be in Base64-encoded PEM format (`.pem`, `.crt`, `.cer`).
9. Return to the Console, and select **Next** to create the connection.
10. Create the connection with PingFederate
 - a. Enter the PingFederate server URL that you copied in the previous step.
 - b. Upload the X.509 signing certificate. The certificate must be in PEM, CER, or CRT format.
11. Select **Create connection**. The system creates the connection in a few seconds.
12. Select **Next**.
13. Select **Test connection** to test your connection. You are directed to a login page for your IdP server. Log in with your IdP credentials. After you log in, go back to the Console to enable the connection.



When using the Console in restricted mode, copy the URL to either an incognito browser window or a separate browser to log in to your IdP.

14. In the Console, select **Next** to review the summary page.
15. Set up notifications.

Choose between seven days or 30 days. The system emails expiry notifications and shows them in the Console to any user with the following roles: Super admin, Org admin, Federation admin, and Federation viewer.

16. Review the federation details and then select **Enable federation**.
17. Select **Finish** to complete the process.

After you enable the federation, users log in to the NetApp Console using their corporate credentials.

Federate with a SAML identity provider

Federate with your SAML 2.0 IdP provider to enable single sign-on (SSO) for the NetApp Console. This allows users to log in using their corporate credentials.

Required role

The Federation admin role is required to make create and manage federations. Federation viewer can view the Federation page. [Learn more about access roles.](#)



You can federate with your corporate IdP or with the NetApp Support Site. You can't federate with both.

NetApp supports service provider-initiated (SP-initiated) SSO only. You need to first configure the identity provider to trust NetApp as a service provider. Then, you can create a connection in the Console that uses the identity provider's configuration.

You can set up a federated connection with your SAML 2.0 provider to enable single sign-on (SSO) for the Console. The process involves configuring your provider to trust NetApp as a service provider and then creating the connection in the Console.

Steps

1. Select **Administration > Identity and access**.
2. Select **Federation** to view the **Federations** page.
3. Select **Configure new federation**.
4. Enter your domain details:
 - a. Choose whether you want to use a verified domain or your email domain. The email domain is the domain associated with the account you are logged in with.
 - b. Enter the name of the federation you are configuring.
 - c. If you choose a verified domain, select the domain from the list.
5. Select **Next**.
6. For your connection method, choose **Protocol** and then select **SAML Identity Provider**.
7. Select **Next**.
8. Configure your SAML identity provider to trust NetApp as a service provider. You need to do this step on your SAML provider server.
 - a. Ensure that your IdP has the attribute `email` set to the user's email address. This is required for the Console to identify users correctly:

```
<saml:AttributeStatement
xmlns:x500="urn:oasis:names:tc:SAML:2.0:profiles:attribute:X500"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <saml:Attribute Name="email"
NameFormat="urn:oasis:names:tc:SAML:1.1:nameid-format:X509SubjectName">
    <saml:AttributeValue
xsi:type="xs:string">email@domain.com</saml:AttributeValue>
  </saml:Attribute>
</saml:AttributeStatement>
```

- a. Use the following values when registering your SAML application with the Console:
 - For the **Reply URL** or **Assertion Consumer Service (ACS) URL**, use <https://netapp-cloud-account.auth0.com/login/callback>
 - For the **Logout URL**, use <https://netapp-cloud-account.auth0.com/logout>

- For **Audience/Entity ID**, use `urn:auth0:netapp-cloud-account:<fed-domain-name-saml>` where `<fed-domain-name-saml>` is the domain name you want to use for federation. For example, if your domain is `example.com`, the Audience/Entity ID would be `urn:auth0:netapp-cloud-account:fed-example-com-samlp`.

b. After creating the trust, copy the following values from your SAML provider server:

- Sign In URL
- Sign Out URL (optional)

c. Download the X.509 certificate from your SAML provider server. It needs to be in PEM, CER, or CRT format.

1. Return to the Console, and select **Next** to create the connection.
2. Create the connection with SAML.

d. Enter the **Sign In URL** of your SAML server.

e. Upload the X.509 certificate that you downloaded from your SAML provider server.

f. Optionally, enter the **Sign Out URL** of your SAML server.

1. Select **Create connection**. The system creates the connection in a few seconds.
2. Select **Next**.
3. Select **Test connection** to test your connection. You are directed to a login page for your IdP server. Log in with your IdP credentials. After you log in, go back to the Console to enable the connection.



When using the Console in restricted mode, copy the URL to either an incognito browser window or a separate browser to log in to your IdP.

1. In the Console, select **Next** to review the summary page.
2. Set up notifications.

Choose between seven days or 30 days. The system emails expiry notifications and shows them in the Console to any user with the following roles: Super admin, Org admin, Federation admin, and Federation viewer.

3. Review the federation details and then select **Enable federation**.
4. Select **Finish** to complete the process.

After you enable the federation, users log in to the NetApp Console using their corporate credentials.

Manage federations

Manage federations in NetApp Console

You can manage your federation in the NetApp Console. You can disable it, update expired credentials, as well as disable it if you no longer need it.

Required roles

The Federation admin role is required to make create and manage federations. Federation viewer can view the Federation page. [Learn more about access roles.](#)

You can also add an additional verified domain to an existing federation, which allows you to use multiple domains for your federated connection.



- If you configured federation using NetApp Cloud Central, import it via the **Federation** page to manage it in the Console. [Learn how to import your federation](#)
- You can view Federation management events such as enabling, disabling, and updating federations on the Audit page. [Learn more about monitoring operations in the NetApp Console.](#)

Enable a federation

If you have created a federation but it is not enabled, you can enable it through the **Federation** page. Enabling a federation allows users associated with the federation to log in to the Console using their corporate credentials. Create and test the federation successfully before enabling it.

Steps

1. Select **Administration > Identity and access**.
2. Select the **Federation** tab.
3. Select the actions menu **...** next to the federation that you want to enable and select **Enable**.

Add a verified domain to an existing federation

You can add a verified domain to an existing federation in the Console to use multiple domains with the same identity provider (IdP).

You must have already verified the domain in the Console before you can add it to a federation. If you haven't verified the domain yet, you can do so by following the steps in [Verify your domain in the Console](#).

Steps

1. Select **Administration > Identity and access**.
2. Select the **Federation** tab.
3. Select the actions menu **⋮** next to the federation that you want to add a verified domain to and select **Update domains**. The **Update domains** dialog box displays the domain already associated with this federation.
4. Select a verified domain from the list of available domains.
5. Select **Update**. New domain users may gain federated Console access within 30 seconds.

Updating an expiring federated connection

You can update the details of a federation in the Console. For example, you'll need to update the federation if the credentials such as a certificate or client secret expire. When needed, update the notification date to remind you to update the connection before it expires.



Update the Console first before updating your IdP to avoid login issues. Stay logged into the Console during the process.

Steps


1. Select **Administration > Identity and access**.
2. Select the **Federation** tab.
3. Select the actions menu (three vertical dots) next to the federation that you want to update and select **Update federation**.

4. Update the details of the federation as needed.
5. Select **Update**.

Test an existing federation

Test the connection of an existing federation to verify that it works. This can help you identify any issues with the federation and troubleshoot them.

Steps


1. Select **Administration > Identity and access**.
2. Select the **Federation** tab.
3. Select the actions menu  next to the federation that you want to add a verified domain to and select **Test connection**.
4. Select **Test**. The system prompts you to log in with your corporate credentials. If the connection is successful, you are redirected to the NetApp Console. If the connection fails, you see an error message indicating the issue with the federation.
5. Select **Done** to return to the **Federation** tab.

Disable a federation

If you no longer need a federation, you can disable it. This prevents users associated with the federation from logging in to the Console using their corporate credentials. You can re-enable the federation later if needed.

Disable a federation before deleting it, such as when decommissioning the IdP or discontinuing federation. This allows you to re-enable it later if needed.

Steps

1. Select **Administration > Identity and access**.
2. Select the **Federation** tab.
3. Select the actions menu  next to the federation that you want to add a verified domain to and select **Disable**.

Delete a federation


If you no longer need a federation, you can delete it. This removes the federation and prevents any users associated with the federation from logging in to the Console using their corporate credentials. For example, if the IdP is being decommissioned or if the federation is no longer needed.

You cannot recover a federation after you delete it. You must create a new federation.



You must disable a federation before you can delete it. You cannot undelete a federation after you delete it.

Steps

1. Select **Administration > Identity and access**.
2. Select **Federations** to view the **Federations** page.
3. Select the actions menu  next to the federation that you want to add a verified domain to and select **Delete**.

Import your federation to NetApp Console

If you have previously set up federation through NetApp Cloud Central (an external application to the NetApp Console) the Federation page prompts you to import your existing federated connection to the Console so you can manage it in the new interface. You can then take advantage of the latest enhancements without having to recreate your federated connection.



After you import your existing federation, you can manage the federation from the **Federations** page. [Learn more about managing federations.](#)

Required role

Organization admin or Federation admin. [Learn more about access roles.](#)

Steps

1. Select **Administration > Identity and access.**
2. Select the **Federation** tab.
3. Select **Import Federation.**

Enforce ONTAP permissions for ONTAP Advanced View (ONTAP System Manager)

By default, the Console agent credentials allow users to access the Advanced View (ONTAP System Manager). You can prompt users for their ONTAP credentials instead. This ensures that a user's ONTAP permissions are applied when they work with ONTAP clusters in both Cloud Volumes ONTAP and ONTAP on-premises clusters.



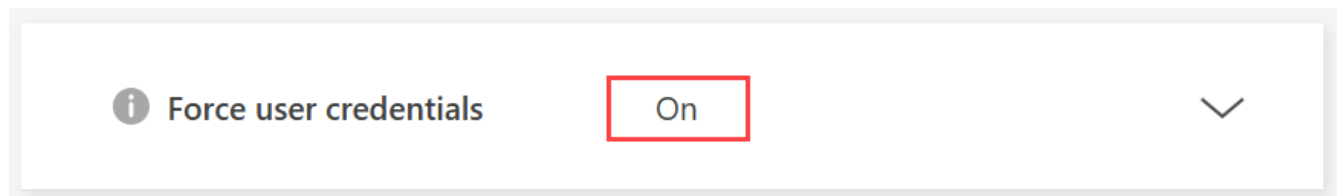
You must have the Organization admin role to edit Console agent settings.

Steps

1. Select **Administration > Agents.**
2. On the **Overview** page, select the action menu for a Console agent and select **Edit agent.**

The Console agent must be active to edit it.

3. Expand the **Force Credentials** option.
4. Select the checkbox to enable the **Force Credentials** option and then select **Save.**
5. Verify that the **Force Credentials** option is enabled.



Enable read-only mode for a NetApp Console organization

As a security precaution, you can enable read-only mode for your NetApp Console

organization. In read-only mode, users can view resources and settings but cannot make any changes.

In read-only mode, users with admin roles must manually elevate their permissions to make changes, which ensures that changes are intentional.

Required access roles

Super admin or Org admin.

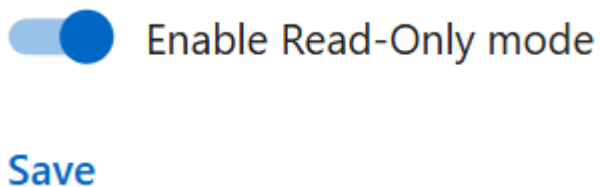
Enable read-only mode for your Console organization

Enable read-only mode to restrict changes to your Console organization. All users can still view resources. Users with admin roles cannot perform any actions in the Console without manually elevating their permissions.

When read-only mode is enabled, users see a banner that notifies them that the organization is in read-only mode. Users must go to User settings to elevate their role.

Steps

1. Select **Administration > Identity and access**.
2. From the **Organizations** tab, select **Edit organization settings** for the organization that you want to set to read-only mode.
3. In the **Read-only mode** section, enable read-only mode by moving the toggle to the **On** position and then select **Save**.



Sign up for NetApp Console as the initial organization administrator

If your company doesn't have a NetApp Console organization, sign up to create one. The first user is the administrator and manages accounts and permissions. You can update roles and add administrators later.

Steps

1. Open a web browser and go to the [NetApp Console](#)
2. If you have a NetApp Support Site account, enter the email address associated with your account directly on the **Log in** page.

The Console signs you up as part of this initial login with your NetApp Support Site credentials.

3. If you want to sign up by creating a Console login, select **Sign up**.
 - a. On the **Sign up** page, enter the required information and select **Next**.



Only English characters are allowed in the sign up form.

- b. Check your inbox for an email from NetApp that includes instructions to verify your email address.

Verify your email address to complete sign up.

4. After you log in, review and accept the End User License Agreement.
5. On the **Welcome** page, create an organization.
6. Select **Let's Start**.

+ As a first-time administrator, follow the guided process to add storage, create a Console agent, and more. [Learn about using the Console Assistant](#).

Next steps

As an administrator, after you complete the steps included in the Console Assistant, you should plan your identity and access strategy, add users to your organization, and assign roles. [Learn about identity and access management for NetApp Console](#)

Sign up or login to NetApp Console when an organization already exists

If your company already has a NetApp Console organization, sign up or log in to access it. Your sign-up or log-in method depends on whether your company uses identity federation or has NetApp Support Site credentials. If not, create a NetApp Console log-in.

Steps

1. Open a web browser and go to the [NetApp Console](#)
2. If you have a NetApp Support Site account or if your company has set up single sign-on (SSO), enter your associated email address or SSO credentials on the **Log in** page. Follow the prompts to complete login.

In both of these cases, you are signed up for the Console as part of this initial login.

3. If you want to sign up by creating a Console login, select **Sign up**.
 - a. On the **Sign up** page, enter the required information and select **Next**.



Only English characters are allowed in the sign up form.

- b. Check your inbox for an email from NetApp that includes instructions to verify your email address.

Verify your email address to complete sign up.

4. After you log in, review and accept the End User License Agreement.
5. If the system prompts you to create an organization, close the dialog box and tell a Console admin so they can add you to your Console organization and give you access. [Learn how to contact an organization administrator](#).

Next steps

After you are given access to your organization, you can start managing storage and using the data services that you are assigned.

Manage organization partnerships

Partnerships in NetApp Console

Creating partnerships between organizations in the NetApp Console lets partners securely manage NetApp resources across organizational boundaries, streamlining collaboration and enhancing security.

Required roles

Partnership admin [Learn more about access roles.](#)

Partnerships allow secure management of NetApp resources across organizations using role-driven relationships in the Console. The initiating organization grants access to its resources, while the accepting organization provides the users or service accounts to be granted access. Partnerships are established through a self-service workflow, giving the initiating organization full control over which resources are shared, what roles are assigned, and the ability to onboard, manage, or revoke partner access as needed.

Customers can authorize MSPs or resellers to manage NetApp environments without requiring complicated setups. Customers can control which clusters partners can access and what roles they have, and can revoke access at any time to maintain security and compliance.

As a partner, you gain centralized visibility and control across customer environments. You can easily switch to a customer's organization to manage resources, run data services, and monitor health within defined boundaries, reducing custom tooling and ensuring alignment with each customer's policies.

1

Assign one or more users the Partnership admin role

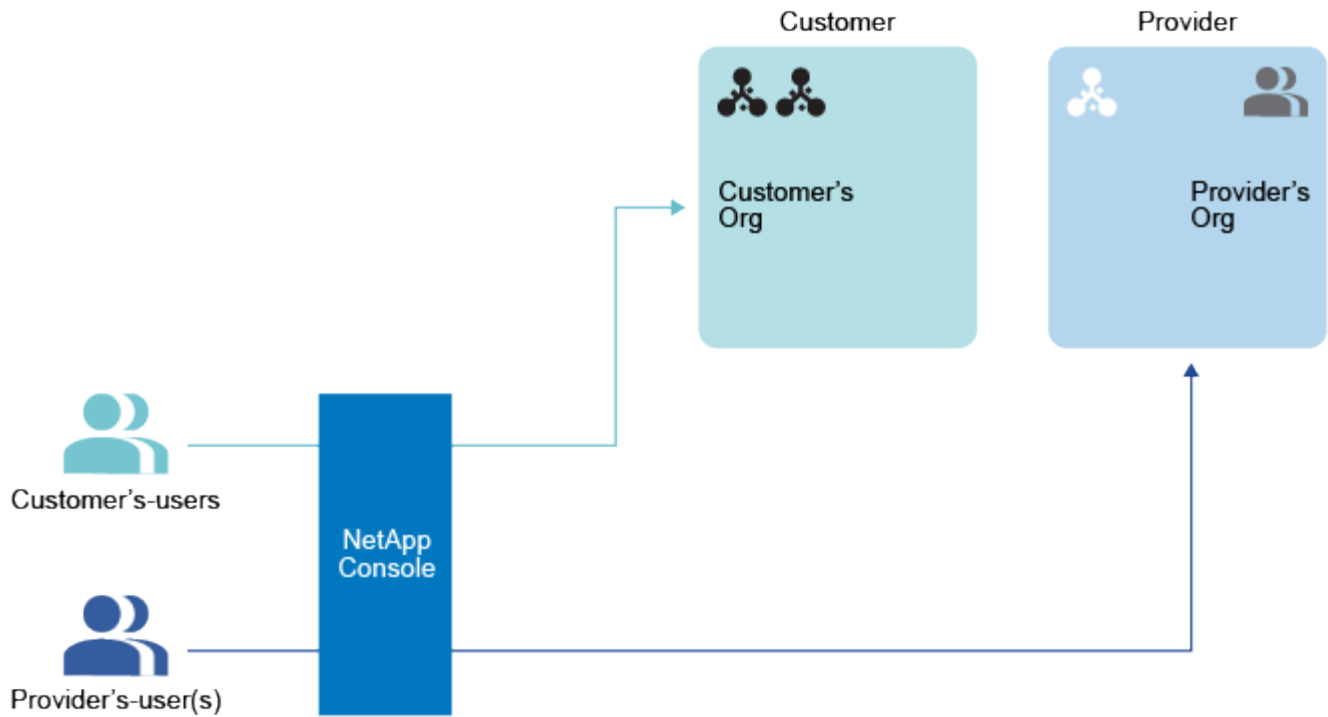
Assign one or more users in both the initiating and receiving organizations the Partnership admin role to create and manage partnerships. You can assign the Partnership viewer role to users who only need to view partnerships, and not manage.

2

Share your organization ID with the initiating organization

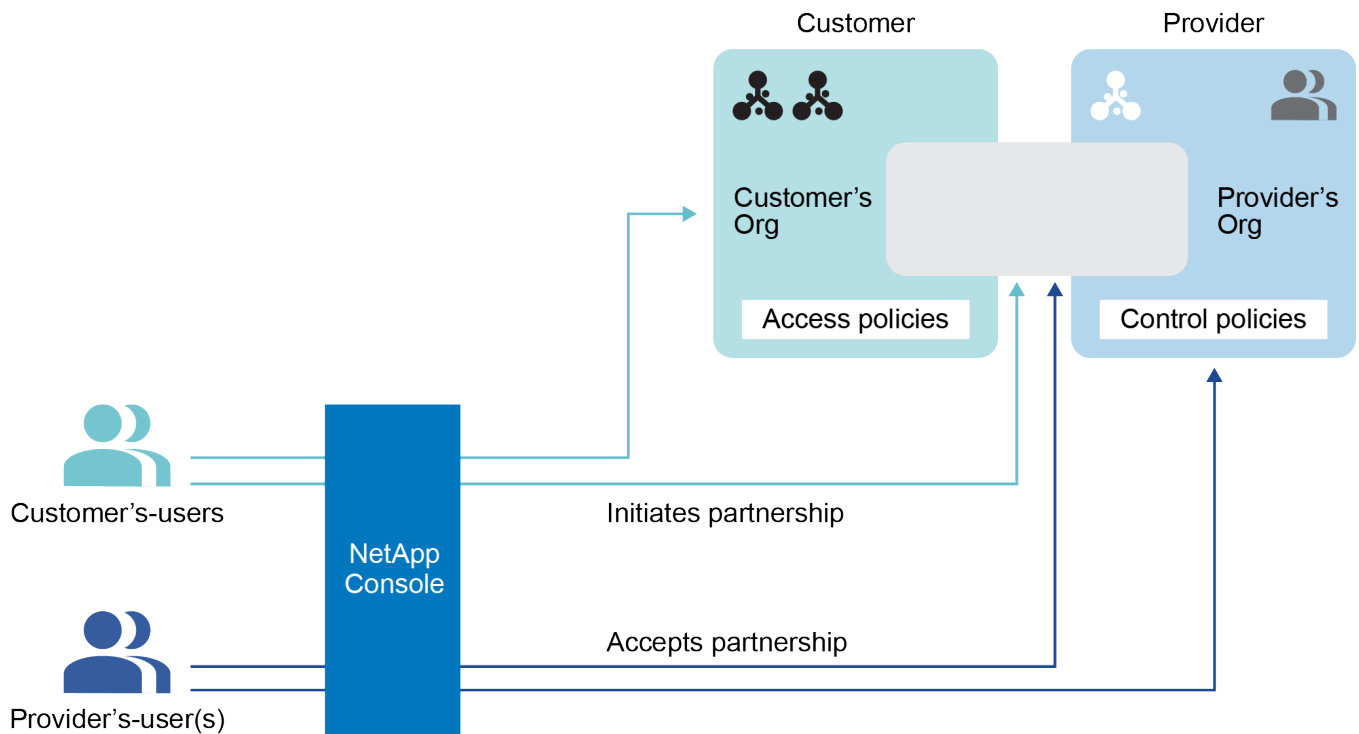
To initiate a partnership, the initiator must know the organization ID of the target organization. Only the respective organization can access this organization ID. Share it directly with the initiating organization outside the NetApp Console via email or another method.

The initiating organization is the organization granting access to its resources.



3 Initiate the partnership within NetApp Console

The organization initiating the partnership does so from within the NetApp Console by sending a partnership request.



4

Approve the partnership

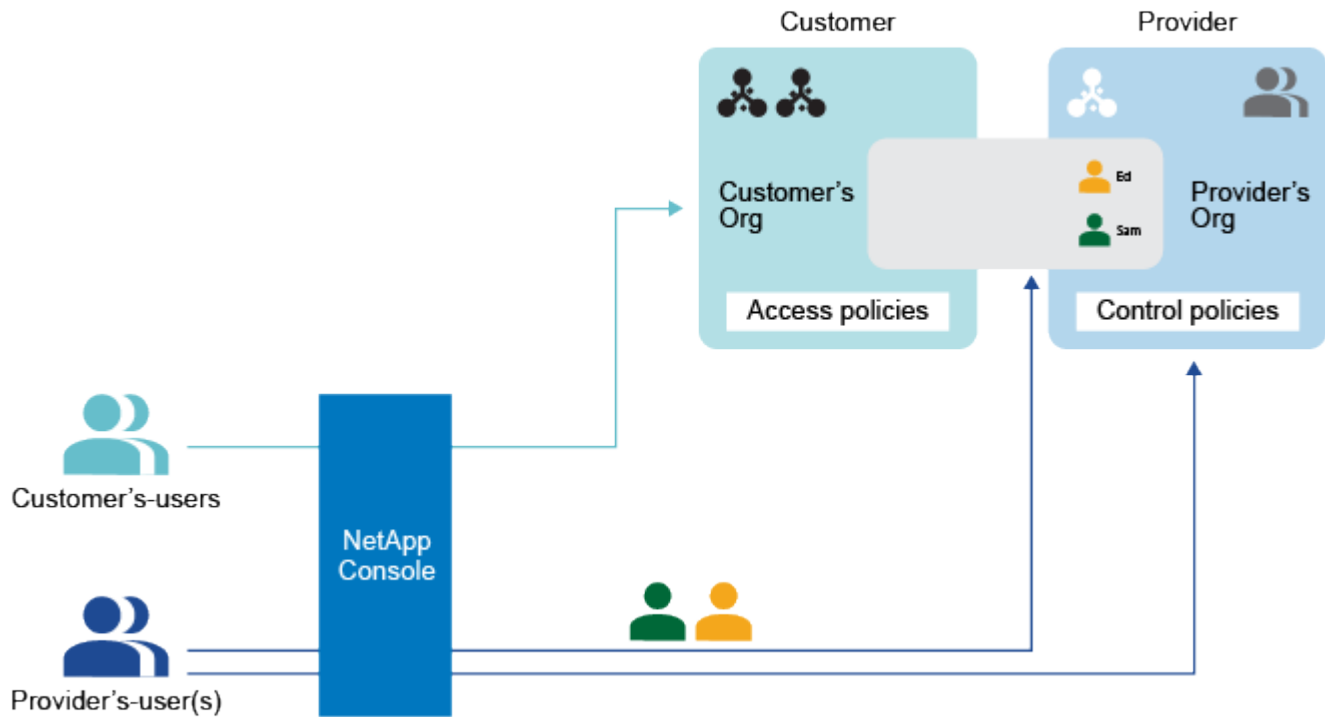
The receiving organization must accept the request.

The receiving organization is the organization being granted access to resources.

5

Assign users to the partnership

The receiving organization assigns specific users or service accounts from your organization to the partnership. The initiating organization assigns roles to these users.

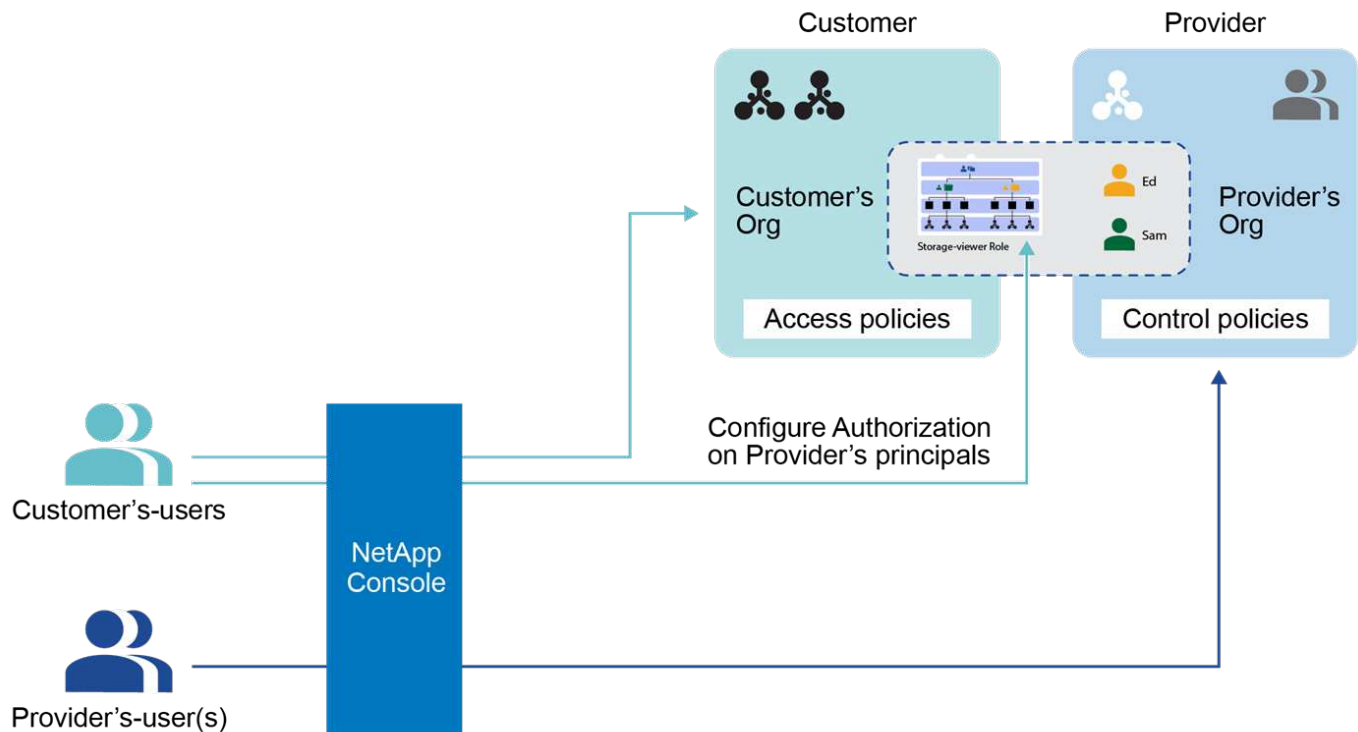


6

Grant assigned users access to resources

If you are the initiating organization, you can grant access to specific resources to the users that were assigned to the partnership. You can revoke access at any time.

You do this by assigning roles for particular projects or folders within your organization.



Manage partnerships in NetApp Console

Create partnerships to establish secure, managed connections between your organization and trusted partners for collaborative NetApp resource management.

Partnerships let you securely manage NetApp resources across boundaries with role-driven relationships in the Console. The initiating organization grants access to its resources, while the accepting organization provides the users or service accounts to be granted access. Partnerships are established through a self-service workflow, giving the initiating organization full control over which resources are shared, what roles are assigned, and the ability to onboard, manage, or revoke partner access as needed.

Required roles

The **Partnership admin** role is required to make create and manage partnerships. The **Partnership viewer** can view the Partnerships page. [Learn more about access roles.](#)

Initiate an organization partnership

You can request a partnership with another organization if you know their organization ID. The receiving organization approves the request before the partnership can proceed.

Before you begin, ensure you have the organization ID of the partner organization and that you have been assigned the **Partnership admin** role.

Steps

1. Select **Administration > Identity and access**.
2. Select the **Partnerships** tab.
3. Select **Add partnership**.
4. In the **Create partnership** dialog box, enter the partner organization ID of the requested partner and select

Add.

The partnership request is sent to the partner organization for approval. You can view the status of the partnership request on the **Partnerships** page.

Approve an organization partnership

An organization partnership request must be accepted by the receiving organization before the partnership can proceed. You must have the **Partnership admin** role to approve and manage partnerships.

Steps

1. Select **Administration > Identity and access**.
2. Select **Partnerships**.
3. Select the **Partnership received** tab.
4. Navigate to the received partnership you want to approve and select **...** and then select **Approve**.
5. Review the details of the partnership, including the name and organization ID of the organization that requested the partnership and select **Next**.
6. Optional, add organization members to the partnership and select **Apply**.

You can add additional members through the **Partnership** page at any time.



Any members you add become visible in the partner's organization where the partner can assign them to resources.

Result

The partnership you approved now shows a status of **Established**. Users with the **Partnership admin** or **Partnership viewer** roles in either organization can view the partnership.

View partnership status

View the status of your partnerships.

Required role

Partnership admin, Partnership viewer. [Learn more about access roles](#).

Steps

1. Select **Administration > Identity and access**.
2. Select the **Partnerships**.
3. Select either the **Initiated partnerships** the **Received partnerships** tab.
4. Review the respective table that displays partnerships and their statuses.

Disable an organization partnership

You must be a member of the initiating organization to disable a partnership. Disabling a partnership immediately revokes access to any resources in your organization that were shared with the partner organization.

Required role

Partnership admin. [Learn more about access roles.](#)

Steps

1. Select **Administration > Identity and access**.
2. Select the **Partnerships**.
3. Select either the **Initiated partnerships** tab.
4. Review the respective table that displays partnerships and their statuses.
5. Navigate to the initiated partnership you want to disable and select **...** and then select **Disable**.

Manage members for a partnership organization

You can add users to a partnership by adding them to the partner organization. After you add users, the partner organization is responsible for assigning them roles for particular resources in their organization.

Required roles

The **Partnership admin** role is required to make create and manage partnerships. The **Partnership viewer** can view the Partnerships page. [Learn more about access roles.](#)

You can remove users from a partnership at any time. Removing a user from a partnership immediately revokes their access to any resources in partner organization.

Add members to a partnership

When you add members to a partnership, the **Partnership admin** of the partner organization must assign them roles for particular resources in their organization before they can access those resources.

After you add members to a partnership, the members display as members in the partner organization where the partner can assign them to resources.

Steps

1. Select **Administration > Identity and access**.
2. Select **Partnerships**.
3. Select the **Partnership received** tab.
4. Select the actions menu **...** next to the established partnership that you want to members and select **Add members**.
5. Choose one or more members to add to to the partnership and select **Add**.

Remove members from a partnership

You can remove members from a partnership at any time. Removing a user from a partnership immediately revokes their access to any resources in partner organization.

If you want to adjust the role that a member has or the resources they can access, the Partnership admin of the partner organization must make those changes.

Steps

1. Select **Administration > Identity and access**.
2. Select **Partnerships**.

3. Select the **Partnership received** tab.
4. Select the actions menu **...** next to member that you want to remove and select **Remove association**.
5. Confirm the action by selecting **Remove** in the dialog box.

View role information for a user

You can view the role that has been assigned to a user and the associated resources.

You cannot change the role associated with a user. If you have questions about the resources or the role provided, contact the administrator of the partner organization.

Steps

1. Select **Administration > Identity and access**.
2. Select **Partnerships**.
3. Select the **Partnership received** tab.
4. From the **Members** page, navigate to a member in the table, select **...** and then select **View details**.
5. In the table, expand the respective row for organization, folder, or project where you want to view the member's assigned role and select the number in the **Role** column.

Provide resource access to partnership users

You can grant access to partnership users by assigning them specific roles for folders and projects within your organization.

Required roles

Partnership admin. [Learn more about access roles](#).

A partner organization must first add members to the partnership before you can assign them roles for resources in your organization. [Learn how to add members to a partnership](#).

Understand roles for partnership users

You can manage roles for members of partner organizations in the same way that you do for your own. However, not all roles are available to partnership users. In particular, you can't grant partner users a role that allows software updates. Updating ONTAP software generally requires direct network access.

You can assign following roles to partner users:

- [Organization admin](#)
- [Folder or project admin](#)
- [Federation admin](#)
- [Federation viewer](#)
- [Backup and recovery admin](#)
- [Backup viewer](#)
- [Restore admin](#)
- [Clone admin](#)
- [Disaster recovery admin](#)

- [Disaster recovery failover admin](#)
- [Disaster recovery application admin](#)
- [Disaster recovery viewer](#)
- [Operations support analyst](#)
- [Classification viewer](#)


[Learn more about predefined roles](#)

Add a role to a partner user


You provide access to your organization's resources by adding a role to a member. When you assign a role, you specify one resource and one role. You can assign more than one role to a user.

For example, if you had two projects and wanted the same user to have the role of Backup and recovery admin for both, you would need to provide the role to the user for each project. Similarly if you wanted to provide a user with two different roles for the same project, you would need to assign each role separately.

Steps

1. Select **Administration > Identity and access**.
2. Select **Partnerships**.
3. Select the **Partnership initiated** tab.
4. Select the actions menu  next to the established partnership that you want view and select **View details**.

The **Member** list displays the members that the partner organization has added to the partnership.

5. Select the actions menu  next to the member that you want to assign a role and select **Add a role**.
6. To add a role, complete the steps in the dialog box:

- **Select an organization, folder, or project:** Choose the level of your resource hierarchy that the member should have permissions for.

If you select the organization or a folder, the member will have permissions to everything that resides within the organization or folder.

- **Select a category:** Choose a role category. [Learn about access roles](#).
- **Select a Role:** Choose a role that provides the member with permissions for the resources that are associated with the organization, folder, or project that you selected.
- **Add role:** If you want to provide access to additional folders or projects within your organization, select **Add role**, specify another folder or project or role category, and then select a role category and a corresponding role.


7. Select **Add new roles**.

Change or remove a role from a partner user



You can change or remove a role that you have assigned to a member of a partner organization.

Steps

1. Select **Administration > Identity and access**.
2. Select **Partnerships**.

3. Select the **Partnership initiated** tab.
4. Select the actions menu  next to the established partnership that you want view and select **View details**.

The **Member** list displays the members that the partner organization has added to the partnership.

5. From the **Members** page, navigate to a member in the table, select  and then select **View details**.
6. In the table, expand the respective row for organization, folder, or project where you want to change the member's assigned role and select **View** in the **Role** column to view the roles assigned to this member.
7. You can change an existing role for a member or remove a role.
 - a. To change a member's role, select **Change** next to the role you want to change. You can only change a role to a role within the same role category. For example, you can change from one data service role to another. Confirm the change.
 - b. To unassign a member's role, select  next to the role to remove the respective role from the member.. You'll be asked to confirm the removal.

Work in a partner organization

Once you have been given a role in a partner organization, you can switch to that organization and perform actions that you have permission to perform.

Use the Organization menu to switch between your organizations and any partner organizations you have access to. [Learn more about switching organizations and projects.](#)

You'll be able to see the resources that have been shared with you in the partner organization and perform actions based on the role that has been assigned to you. Work with your partnership admin to ensure you have the appropriate role for the resources you need to access.

Monitor NetApp Console operations

You can monitor the status of the operations that the Console is performing to see if there are any issues that you need to address. You can view the status from the Audit page, the Notification Center, or have notifications sent to your email.

The table highlights the features of the Audit page and Notification Center by comparing them.

Notification Center	Audit page
Shows high level status for events and actions	Provides details for each event or action for further investigation
Shows status for the current login session (the information does not appear in the Notification Center after you log off)	Retains status for the last month
Shows only actions initiated in the user interface	Shows all actions from the UI or APIs
Shows user-initiated actions	Shows all actions, whether user-initiated or system-initiated
Filter results by importance	Filter by service, action, user, status, and more

Notification Center	Audit page
Provides the ability to email notifications to users and to others	No email capability

Audit user activity from the Audit page

Use the Audit page to identify who performed an action or its status.

The Audit page shows the actions that users completed to manage your organization or account. This includes management actions such as associating users, creating systems, creating agents, and more.

You can also verify who added a member to an organization or that a project was deleted successfully.

Steps

1. Select **Administration > Audit**.
2. Use the filters above the table to change which actions display in the table.

For example, you can use the **Service** filter to show actions related to a specific service, or you can use the **User** filter to show actions related to a specific user account.

Download audit logs from the Audit page


You can download the audit logs from the Audit page to a CSV file. This enables you to keep a record of the actions that users perform in your organization. The CSV file includes all columns in the downloaded CSV file, regardless of filters or displayed columns on the Audit page.

Steps

1. In the **Audit** page, select the download icon in the upper right corner of the table.

Monitor activities using the Notification Center

Notifications track Console operations to confirm success. They enable you to view the status for many Console actions that you initiated during your current login session. Not all Console services report information into the Notification Center.

You can display the notifications by selecting the notification bell () in the menu bar. The color of the little bubble in the bell indicates the highest level severity notification that is active. So if you see a red bubble, it means there's an important notification that you should look at.

You can also configure the Console to send certain types of notifications by email so you can be informed of important system activity even when you're not logged into the system. Emails can be sent to any users who are part of your organization, or to any other recipients who need to be aware of certain types of system activity. See how to [set email notification settings](#).

Comparing the Notification Center with alerts

The Notification Center enables you to view the status of operations you've initiated and set up alert notifications for certain types of system activities. Meanwhile, alerts enable you to view issues or potential risks in your ONTAP storage environment related to capacity, availability, performance, protection, and security.

[Learn more about NetApp Console alerts](#)

Notification types

The Console classifies notifications into the following categories:

Notification type	Description
Critical	A problem occurred that might lead to service disruption if corrective action is not taken immediately.
Error	An action or process ended with failure, or could lead to failure if corrective action is not taken.
Warning	An issue that you should be aware of to make sure it does not reach the critical severity. Notifications of this severity do not cause service disruption, and immediate corrective action might not be required.
Recommendation	A system recommendation for you to take an action to improve the system or a certain service; for example: costs saving, suggestion for new services, recommended security configuration, etc.
Information	A message that provides additional information about an action or process.
Success	An action or process completed successfully.

Filter notifications

By default you'll see all active notifications in the Notification Center. You can filter the notifications that you see to show only those notifications that are important to you. You can filter by "Service" and by notification "Type".

The image shows a user interface for filtering notifications. It consists of two side-by-side panels, each with a title and a list of items with checkboxes. The left panel is titled 'Filter Services (All)' and contains three items: 'Digital Wallet (3)' with a checked checkbox, 'Active IQ (2)' with a checked checkbox, and 'AppTemplate (1)' with an unchecked checkbox. Below the list are two buttons: 'Clear' and 'Apply'. The right panel is titled 'Filter Type (All)' and contains six items: 'Information (0)' with an unchecked checkbox, 'Success (1)' with an unchecked checkbox, 'Warning (2)' with a checked checkbox, 'Error (1)' with a checked checkbox, 'Critical (0)' with a checked checkbox, and 'Recommendation (0)' with an unchecked checkbox. Below the list are two buttons: 'Clear' and 'Apply'.

For example, if you want to see only "Error" and "Warning" notifications for Console operations, select those entries and you'll see only those types of notifications.

Dismiss notifications

You can remove notifications from the page if you no longer need to see them. You can dismiss notifications individually or all at once.

To dismiss all notifications, in the Notification Center, select  and select **Dismiss All**.

To dismiss individual notifications, hover your cursor over the notification and select **Dismiss**.

Set email notification settings

You can send specific types of notifications by email so you can be informed of important system activity even when you're not logged in. Emails can be sent to any users who are part of your organization or account, or to any other recipients who need to be aware of certain types of system activity.



- The Console sends email notifications for the agent, licenses and subscriptions, NetApp Copy and Sync, and NetApp Backup and Recovery.
- Sending email notifications is not supported when the Console agent is installed in a site without internet access.

The filters you set in the Notification Center do not determine the types of notifications you receive by email. By default, any Organization admin will receive emails for all "Critical" and "Recommendation" notifications. These notifications are across all services - you can't choose to receive notifications for only certain services, for example agents or NetApp Backup and Recovery.

All other users and recipients are configured not to receive any notification emails - so you'll need to configure notification settings for any additional users.

You must have the Organization admin role to customize the notifications settings.

Steps

1. Select **Administration > Notifications settings**.
2. Select **Organization users** or **Additional recipients**.

The **Additional recipients** page allows you to configure the Console to notify people who are members of your Console organization.

3. Select a user, or multiple users, from either the *Organization users* page or the *Additional Recipients* page, and choose the type of notifications to be sent:
 - To make changes for a single user, select the menu in the Notifications column for that user, check the types of Notifications to be sent, and select **Apply**.
 - To make changes for multiple users, check the box for each user, select **Manage Email Notifications**, check the types of Notifications to be sent, and select **Apply**.

Add additional email recipients

The users who appear in the *Organization users* page are populated automatically from the users in your organization or account. You can add email addresses in the *Additional Recipients* page for other people, or groups, who do not have access to the Console, but who need to be notified about certain types of alerts and notifications.

Steps

1. From the **Notifications settings** page, select **Add New Recipients**.

Add New Recipient

Email

saul.jenkin@gmail.com

Name

Saul Jenkin

Notification Type

Critical ×

Recommendation ×

Error ×

×

Add New Recipient

Cancel

2. Enter the name, email address, and select the types of notifications that recipient will receive, and select **Add New Recipient**.

Copyright information

Copyright © 2026 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.

LIMITED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (b)(3) of the Rights in Technical Data -Noncommercial Items at DFARS 252.227-7013 (FEB 2014) and FAR 52.227-19 (DEC 2007).

Data contained herein pertains to a commercial product and/or commercial service (as defined in FAR 2.101) and is proprietary to NetApp, Inc. All NetApp technical data and computer software provided under this Agreement is commercial in nature and developed solely at private expense. The U.S. Government has a non-exclusive, non-transferrable, nonsublicensable, worldwide, limited irrevocable license to use the Data only in connection with and in support of the U.S. Government contract under which the Data was delivered. Except as provided herein, the Data may not be used, disclosed, reproduced, modified, performed, or displayed without the prior written approval of NetApp, Inc. United States Government license rights for the Department of Defense are limited to those rights identified in DFARS clause 252.227-7015(b) (FEB 2014).

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.