Get started with NetApp Console (restricted mode)

NetApp Console setup and administration

NetApp
January 27, 2026

Table of Contents

Get started with NetApp Console (restricted mode)

Getting started workflow (restricted mode)

Get started with the NetApp Console in restricted mode by preparing your environment and deploying the Console agent.

Restricted mode is typically used by state and local governments and regulated companies, including deployments in AWS GovCloud and Azure Government regions. Before getting started, ensure you have an understanding of Console agents and deployment modes.

1 Prepare for deployment

a. Prepare a dedicated Linux host that meets requirements for CPU, RAM, disk space, container orchestration tool, and more.

b. Set up networking that provides access to the target networks, outbound internet access for manual installations, and outbound internet for day-to-day access.

c. Set up permissions in your cloud provider so that you can associate those permissions with the Console agent instance after you deploy it.

2 Deploy the Console agent

a. Install the Console agent from your cloud provider's marketplace or by manually installing the software on your own Linux host.

b. Set up the NetApp Console by opening a web browser and entering the Linux host's IP address.

c. Provide the Console agent with the permissions that you previously set up.

3 Subscribe to NetApp Intelligent Services (optional)

Optional: Subscribe to NetApp Intelligent Services from your cloud provider's marketplace to pay for data services at an hourly rate (PAYGO) or through an annual contract. NetApp Intelligent Services include NetApp Backup and Recovery, Cloud Volumes ONTAP, NetApp Cloud Tiering, NetApp Ransomware Resilience and NetApp Disaster Recovery. NetApp Data Classification is included with your subscription at no additional cost.

Prepare for deployment in restricted mode

Prepare your environment before you deploy NetApp Console in restricted mode. You need to review host requirements, prepare networking, set up permissions, and more.

Step 1: Understand how restricted mode works

Understand how the NetApp Console works in restricted mode before starting.

Use the browser-based interface available locally from the installed NetApp Console agent. You can't access the NetApp Console from the web-based console that's provided through the SaaS layer.

In addition, not all Console features and NetApp data services are available.

Learn how restricted mode works.

Step 2: Review installation options

In restricted mode, you can only install the Console agent in the cloud. The following installation options are available:

- From the AWS Marketplace
- From the Azure Marketplace
- Manually installing the Console agent on your own Linux host running in AWS, Azure, or Google Cloud

Step 3: Review host requirements

A host must meet specific OS, RAM, and port requirements to run the Console agent.

When you deploy the Console agent from the AWS or Azure Marketplace, the image includes the required OS and software components. You simply need to choose an instance type that meets CPU and RAM requirements.

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.

Azure VM size

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

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				
	9.6 • 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

Operating system	Supported OS versions	Supported agent versions	Required container tool	SELinux
	9.1 to 9.4 • 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
	8.6 to 8.10 • 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

Operating system	Supported OS versions	Supported agent versions	Required container tool	SELinux
	22.04 LTS	3.9.50 or later	Docker Engine 23.0.6 to 28.0.0.	Not supported

Step 4: Install Podman or Docker Engine

To manually install the Console agent, prepare the host by installing 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 5: Prepare network access

Set up network access so the Console agent can manage resources in your public cloud. In addition to having a virtual network and subnet for the Console agent, you need to ensure that the following requirements are met.

Connections to target networks

Ensure the Console agent has a network connection to the storage locations. For example, the VPC or VNet where you plan to deploy Cloud Volumes ONTAP, or the data center where your on-premises ONTAP clusters reside.

Prepare networking for user access to NetApp Console

In restricted mode, users access the Console from the Console agent VM. The Console agent contacts a few endpoints to complete data management tasks. These endpoints are contacted from a user's computer when completing specific actions from the Console.

Console agents previous to version 4.0.0 need additional endpoints. If you upgraded to 4.0.0 or later, you can remove the old endpoints from your allow list. Learn more about the required network access for versions previous to 4.0.0.

+

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://cdn.auth0.com https://services.cloud.netapp.com	Your web browser connects to these endpoints for centralized user authentication through the NetApp Console.

Outbound internet access for day-to-day operations

The Console agent's network location must have outbound internet access. It needs to be able to reach the SaaS services of the NetApp Console as well as endpoints within your respective public cloud environment.

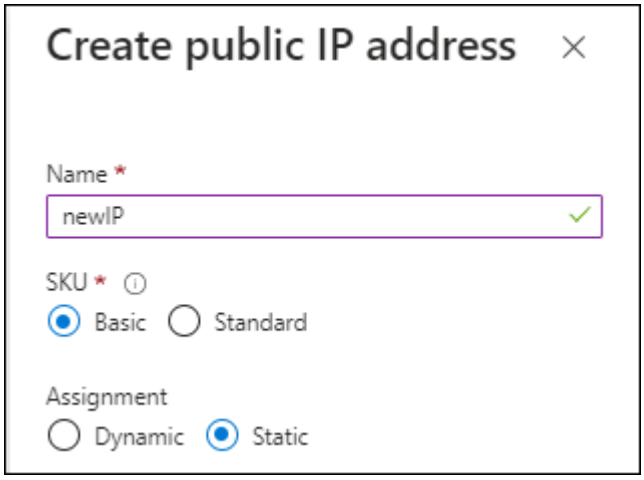
Endpoints	Purpose
AWS environments	
AWS services (amazonaws.com): • CloudFormation • Elastic Compute Cloud (EC2) • Identity and Access Management (IAM) • Key Management Service (KMS) • Security Token Service (STS) • Simple Storage Service (S3)	To manage AWS resources. The endpoint depends on your AWS region. Refer to AWS documentation for details
Amazon FsX for NetApp ONTAP: • 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
Azure environments	
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.usgovcloudapi.net https://login.microsoftonline.us https://blob.core.usgovcloudapi.net https://core.usgovcloudapi.net	To manage resources in Azure Government 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.
Google Cloud environments	
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.
NetApp Console endpoints	
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. • 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. 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. • When you update to the current endpoints in your firewall, your existing agents will continue to work.

Public IP address in Azure

If you want to use a public IP address with the Console agent VM in Azure, the IP address must use a Basic SKU to ensure that the Console uses this public IP address.



If you use a Standard SKU IP address instead, then the Console uses the *private* IP address of the Console agent, instead of the public IP. If the machine that you're using to access the Console doesn't have access to that private IP address, then actions from the Console will fail.

Azure documentation: Public IP SKU

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

If you're planning to create a Console agent from your cloud provider's marketplace, implement this networking requirement after you create the Console agent.

Step 6: Prepare cloud permissions

The Console agent requires permissions from your cloud provider to deploy Cloud Volumes ONTAP in a virtual network and to use NetApp data services. You need to set up permissions in your cloud provider and then associate those permissions with the Console agent.

To view the required steps, choose the authentication option to use for your cloud provider.

AWS IAM role

Use an IAM role to provide the Console agent with permissions.

If you're creating the Console agent from the AWS Marketplace, you are prompted to select that IAM role when you launch the EC2 instance.

If you're manually installing the Console agent on your own Linux host, attach the role to the EC2 instance.

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.
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 for the Console agent EC2 instance.

AWS access key

Set up permissions and an access key for an IAM user. You'll need to provide the Console with the AWS access key after you install the Console agent and set up the Console.

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.

Azure role

Create an Azure custom role with the required permissions. You'll assign this role to the Console agent VM.

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

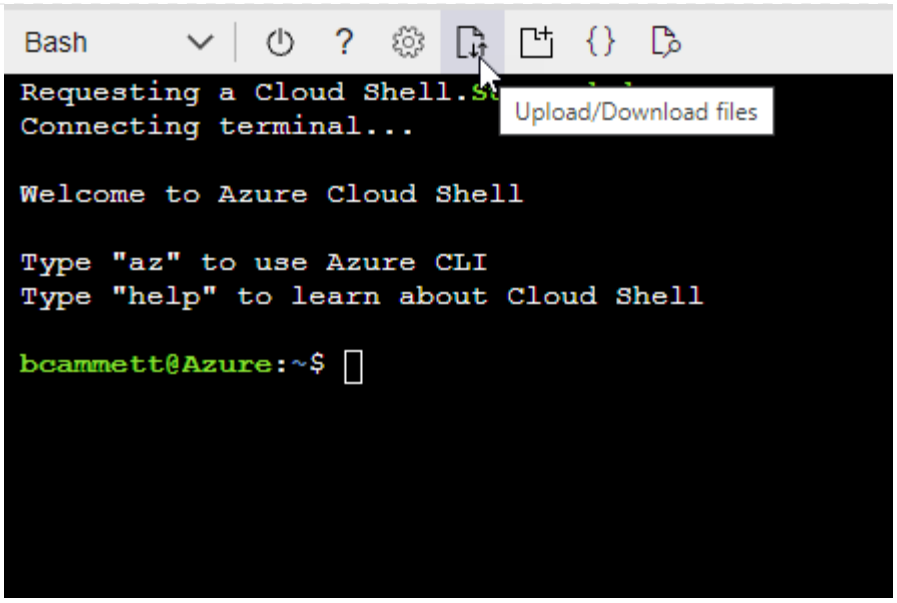
   ```
   "AssignableScopes": [
   "/subscriptions/d333af45-0d07-4154-943d-c25fbzzzzzzz",
   "/subscriptions/54b91999-b3e6-4599-908e-416e0zzzzzzz",
   "/subscriptions/398e471c-3b42-4ae7-9b59-ce5bbzzzzzzz"
   ]
   ```

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
```

Azure service principal

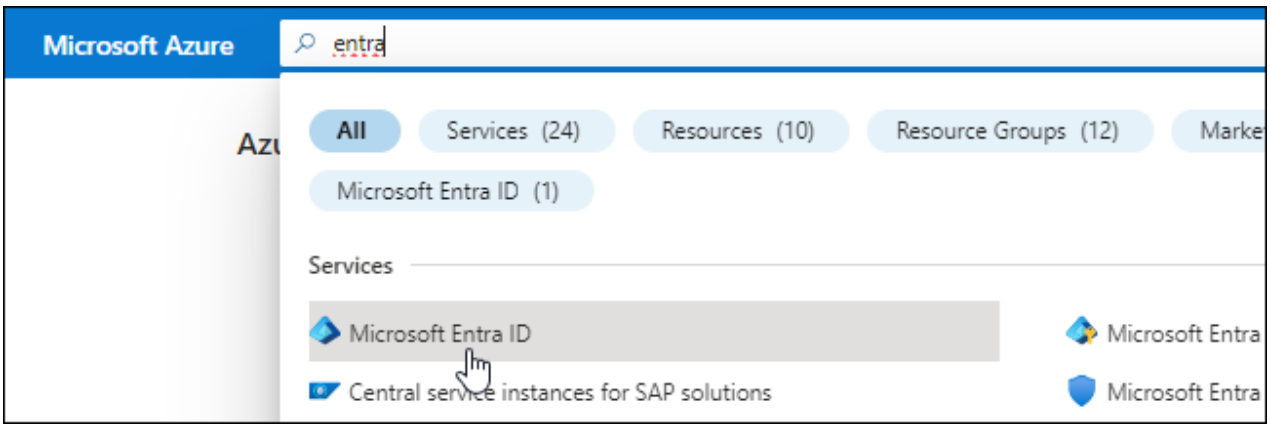
Create and set up a service principal in Microsoft Entra ID and obtain the Azure credentials that the Console needs. You need to provide the Console with these credentials after you install the Console agent.

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.

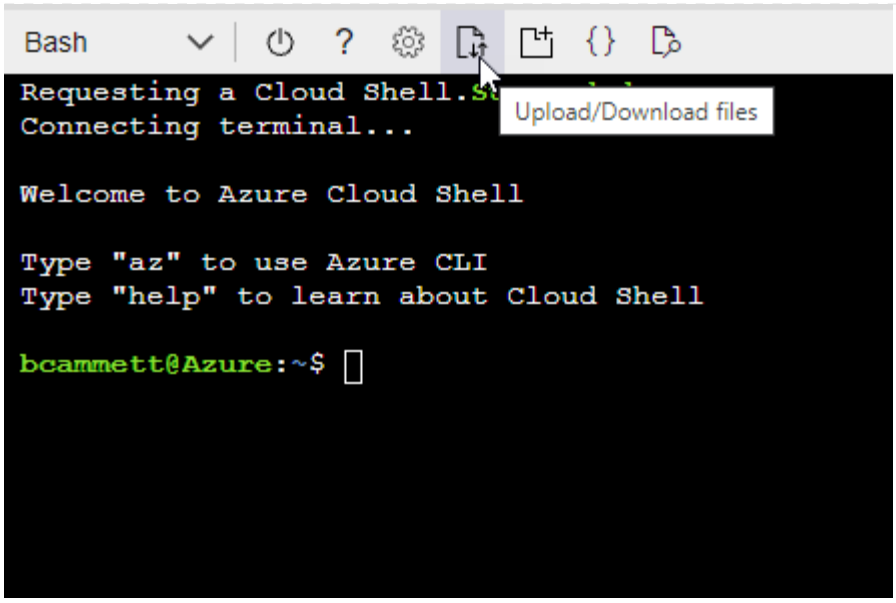
      Example

      ```
      "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.

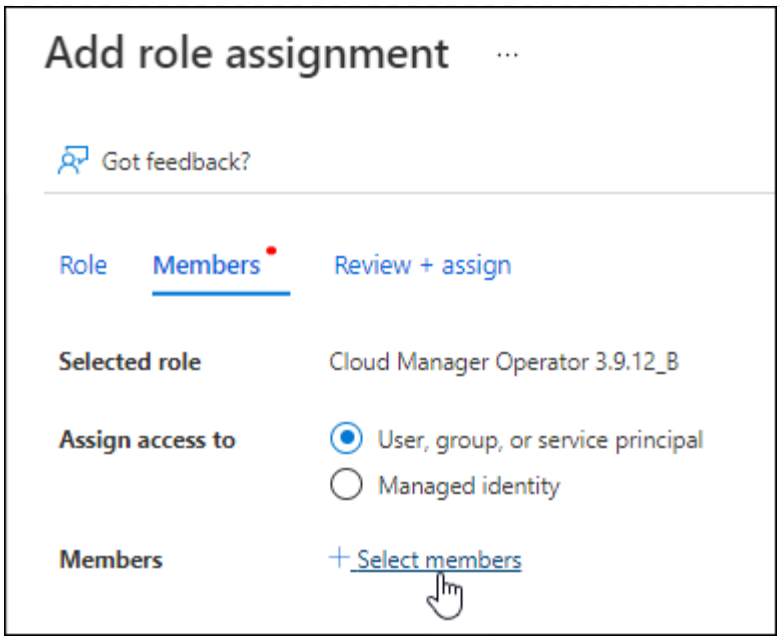


- Use the Azure CLI to create the custom role:

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

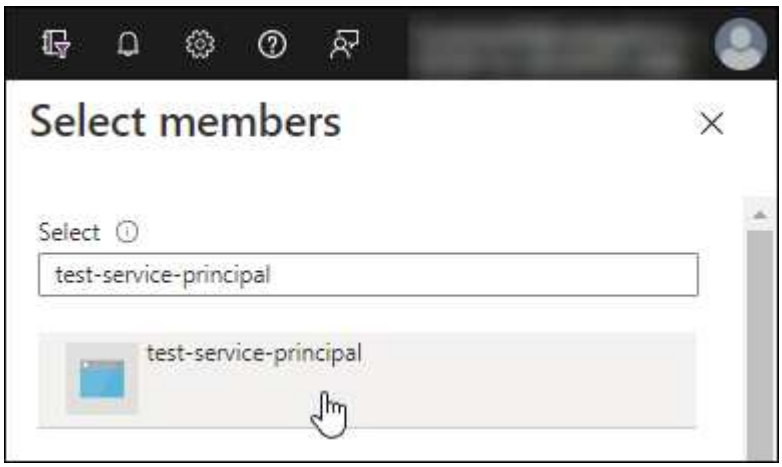
You should now have a custom role called Console Operator that you can assign to the Console agent virtual machine.

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**.

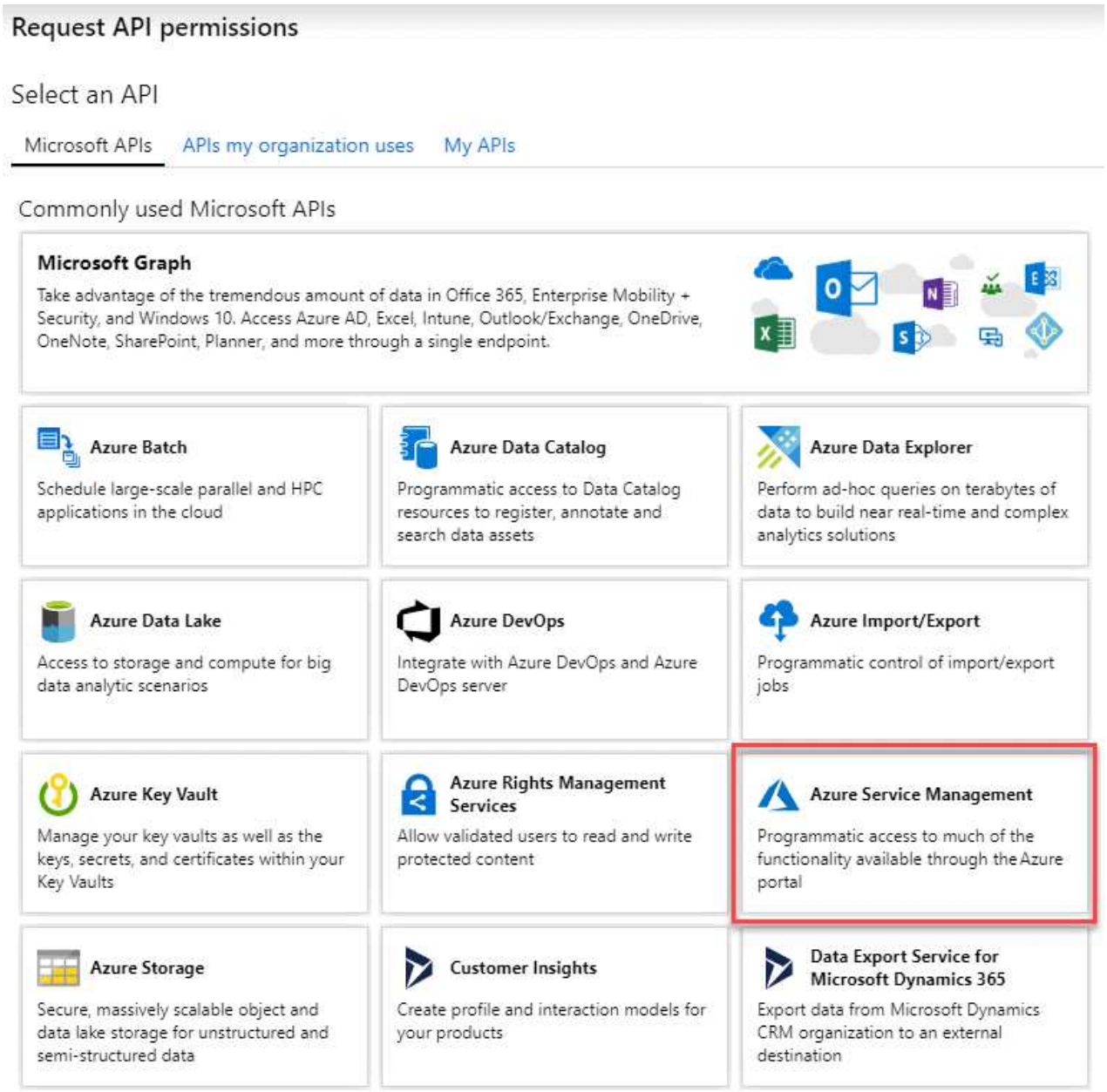
f. Select **Review + assign**.

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

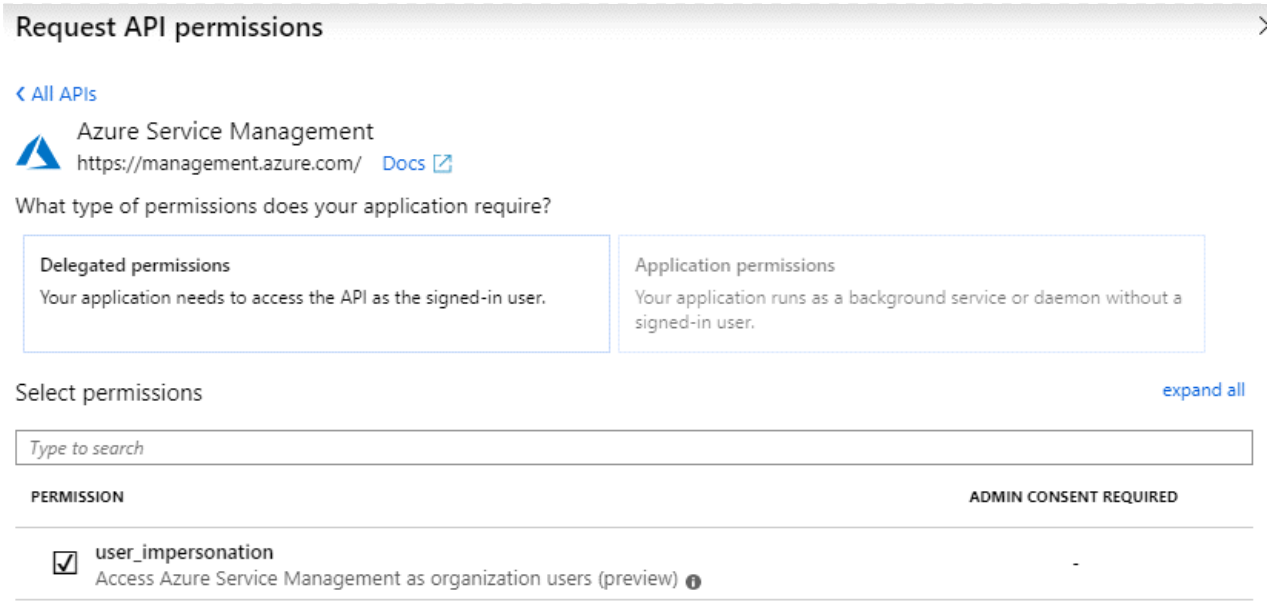
If you want to deploy Cloud Volumes ONTAP from multiple Azure subscriptions, then you must bind the service principal to each of those subscriptions. In the NetApp Console, you can select the subscription that you want to use when deploying Cloud Volumes ONTAP.

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**.

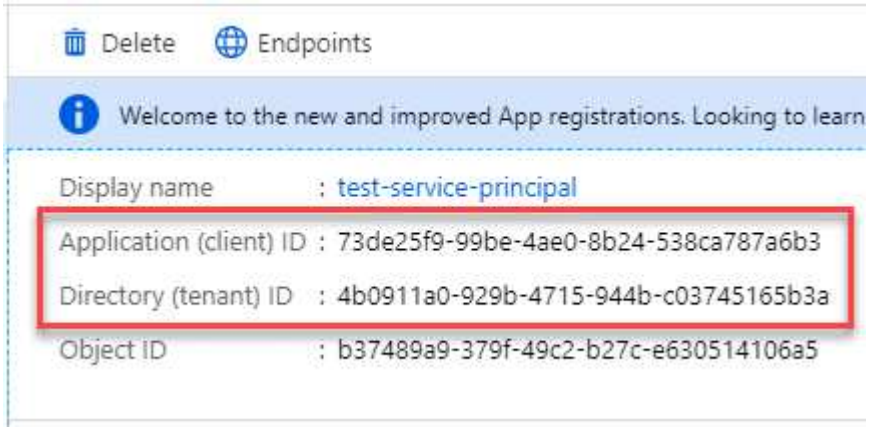


4. Select **Access Azure Service Management as organization users** and then select **Add permissions**.



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.

Result

Your service principal is now set up 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.

Google Cloud service account

Create a role and apply it to a service account that you'll use for the Console agent VM instance.

Steps

1. Create a custom role in Google Cloud:
   a. Create a YAML file that includes the permissions defined in the Console agent policy for Google Cloud.
   b. From Google Cloud, activate cloud shell.
   c. Upload the YAML file that includes the required permissions for the Console agent.
   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 agent --project=myproject
      --file=agent.yaml
      ```

      Google Cloud docs: Creating and managing custom roles

2. Create a service account in Google Cloud:
   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

Step 7: Enable Google Cloud APIs

Several APIs are required to deploy Cloud Volumes ONTAP 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))

Deploy the Console agent in restricted mode

Deploy the Console agent in restricted mode so that you can use the NetApp Console with limited outbound connectivity. To get started, install the Console agent, set up the Console by accessing the user interface that's running on the Console agent, and then provide the cloud permissions that you previously set up.

Step 1: Install the Console agent

Install the Console agent from your cloud provider's marketplace or manually on a Linux host.

You need to have prepared your environment before you install the Console agent. You can install from the AWS Marketplace, from the Azure Marketplace, or manually on your own Linux host running in AWS, Azure, or Google Cloud.

AWS Commercial Marketplace

Before you begin

Have the following:

- A VPC and subnet that meets networking requirements.

  Learn about networking requirements

- An IAM role with an attached policy that includes the required permissions for the Console agent.

  Learn how to set up AWS permissions

- Permissions to subscribe and unsubscribe from the AWS Marketplace for your IAM user.
- An understanding of CPU and RAM requirements for the agent.

  Review agent requirements.

- 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**.

Result

AWS launches the software with the specified settings. The Console agent deploys in approximately five minutes.

What's next?

Set up the NetApp Console.

AWS Gov Marketplace

Before you begin

Have the following:

- A VPC and subnet that meets networking requirements.

  Learn about networking requirements

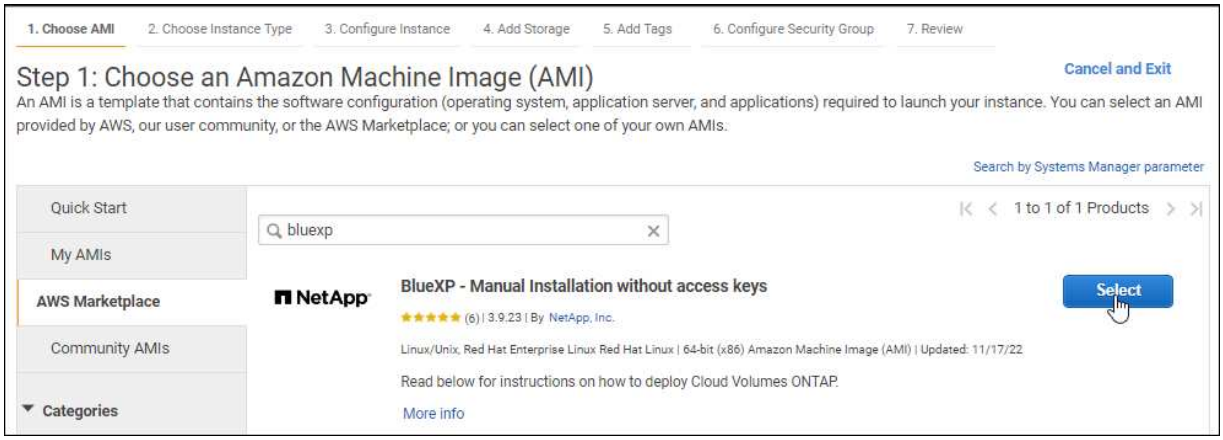
- An IAM role with an attached policy that includes the required permissions for the Console agent.

  Learn how to set up AWS permissions

- Permissions to subscribe and unsubscribe from the AWS Marketplace for your IAM user.
- A key pair for the EC2 instance.

Steps

1. Go to the NetApp Console agent offering in the AWS Marketplace.
   a. Open the EC2 service and select **Launch instance**.
   b. Select **AWS Marketplace**.
   c. Search for NetApp Console and select the offering.

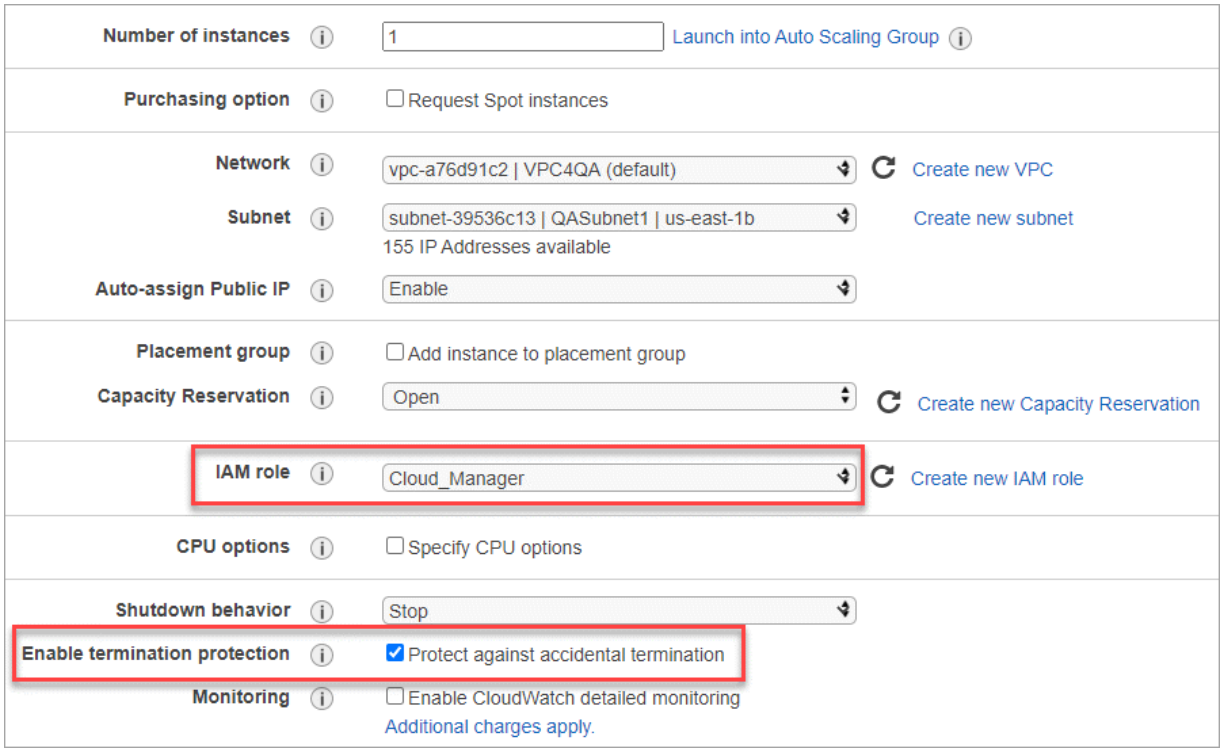


   d. Select **Continue**.

2. Follow the prompts to set up and start the instance:
   - **Choose an Instance Type**: Depending on region availability, choose one of the supported instance types (t3.2xlarge is recommended).

     Review the instance requirements.

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



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

Result

AWS launches the software with the specified settings. The Console agent deploys in approximately five minutes.

What's next?

Set up the Console.

Azure Gov Marketplace

Before you begin

You should have the following:

- A VNet and subnet that meets networking requirements.

Learn about networking requirements

- An Azure custom role that includes the required permissions for the Console agent.

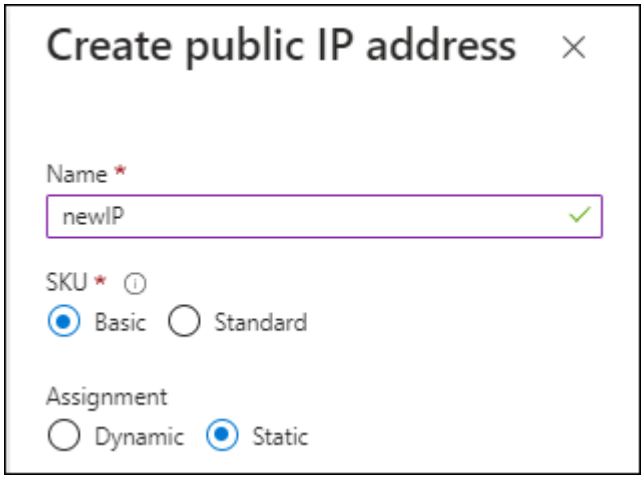
  Learn how to set up Azure permissions

Steps

1. Go to the NetApp Console agent VM page in the Azure Marketplace.
   - Azure Marketplace page for commercial regions
   - Azure Marketplace page for Azure Government 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.
   - **Public IP**: To use a public IP address with the Console agent VM, select a Basic SKU.



   If you use a Standard SKU IP address instead, then the Console uses the *private* IP address of the Console agent, instead of the public IP. If the machine you use to access the Console cannot reach the private IP address, the Console does not work.

   Azure documentation: Public IP SKU

   - **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**.

     A managed identity lets the Console agent VM identify itself to Microsoft Entra ID without 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.

Result

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

What's next?

Set up the NetApp Console.

Manual install (must use for Google Cloud)

You can install the Console agent manually on your own Linux host running in AWS, Azure, or Google Cloud.

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.

- You need to disable the configuration check that verifies outbound connectivity during installation. The manual install fails if this check is not disabled. Learn how to disable configuration checks for manual installations.
- Depending on your operating system, either Podman or Docker Engine is required before you install the Console agent.

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.

Result

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

What's next?

Set up the NetApp Console.

Step 2: Set up NetApp Console

When you access the console for the first time, you are prompted to choose an organization for the Console agent and need to enable restricted mode.

Before you begin

The person who sets up the Console agent must log in to the Console using a login that doesn't already belong

to a Console organization.

If your login is associated with another organization, you need to sign up with a new login. Otherwise, you do not see the option to enable restricted mode on the setup screen.

Steps

1. Open a web browser from a host that has a connection to the Console agent instance and enter the following URL of the Console agent you installed.
2. Sign up or log in to the NetApp Console.
3. After you're logged in, set up the Console:
   a. Enter a name for the Console agent.
   b. Enter a name for a new Console organization.
   c. Select **Are you running in a secured environment?**
   d. Select **Enable restricted mode on this account**.

      Note that you can't change this setting after the account is created. You can't enable restricted mode later and you can't disable it later.

      If you deployed the Console agent in a Government region, the checkbox is already enabled and can't be changed. This is because restricted mode is the only mode supported in Government regions.

   e. Select **Let's start**.

Result

The Console agent is now installed and set up with your Console organization. All users need to access the Console using the IP address of the Console agent instance.

What's next?

Provide the Console with the permissions that you previously set up.

Step 3: Provide permissions to the Console agent

If you installed the Console agent from the Azure Marketplace or manually, you need to give the permissions you set up earlier.

These steps don't apply if you deployed the Console agent from the AWS Marketplace because you chose the required IAM role during deployment.

Learn how to prepare cloud permissions.

AWS IAM role

Attach the IAM role that you previously created to the EC2 instance where you installed the Console agent.

These steps apply only if you manually installed the Console agent in AWS. For AWS Marketplace deployments, you already associated the Console agent instance with an IAM role that includes the required permissions.

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**.

AWS access key

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

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**.

Azure 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.

Azure service principal

Provide the NetApp Console with the credentials for the Azure service principal that you previously setup.

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

Google Cloud service account

Associate the service account with the Console agent VM.

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 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.

Subscribe to NetApp Intelligent Services (restricted mode)

Subscribe to NetApp Intelligent Services from your cloud provider's marketplace to pay for data services at an hourly rate (PAYGO) or through an annual contract. If you

purchased a license from NetApp (BYOL), you also need to subscribe to the marketplace offering. Your license is always charged first, but you'll be charged at the hourly rate if you exceed your licensed capacity or if the license's term expires.

A marketplace subscription enables charging for the following data services with restricted mode:

- NetApp Backup and Recovery
- Cloud Volumes ONTAP
- NetApp Cloud Tiering
- NetApp Ransomware Resilience
- NetApp Disaster Recovery

NetApp Data Classification is enabled through your subscription, but there is no charge for using classification.

Before you begin

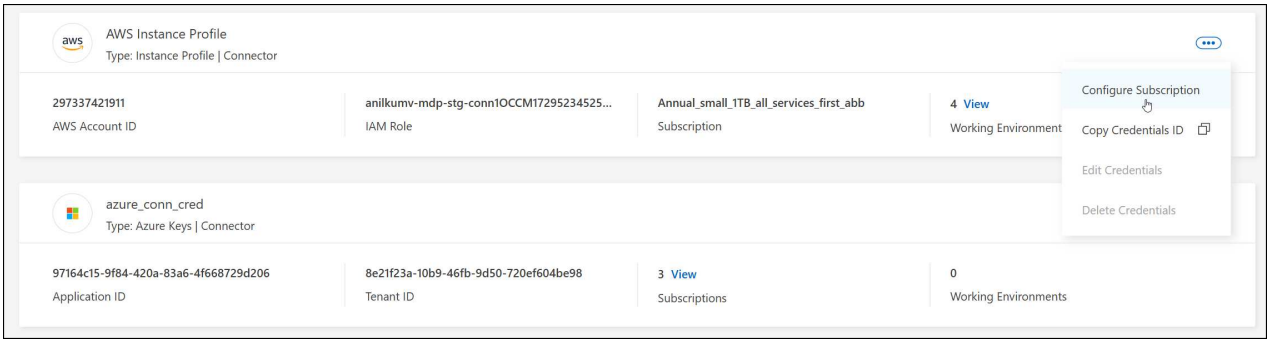
You must have already deployed a Console agent in order to subscribe to data services. You need to associate a marketplace subscription to the cloud credentials connected to a Console agent.

AWS

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**.

Azure

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.

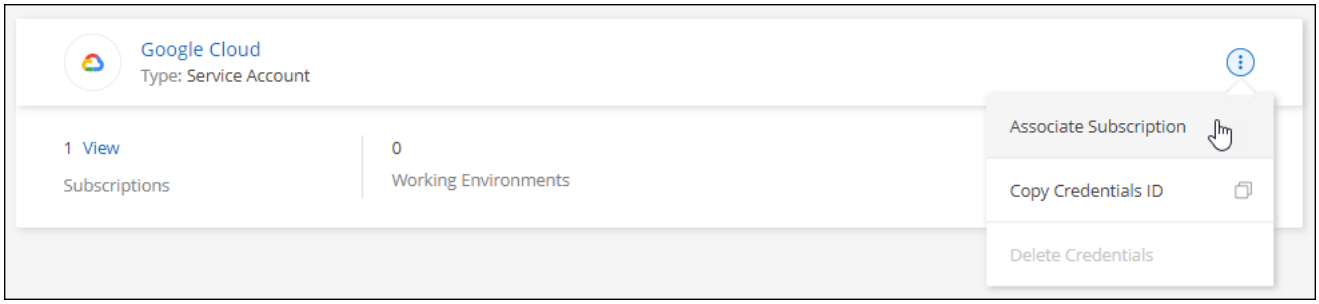
        For all other organizations or accounts, you'll need to manually associate the subscription by repeating these steps.

      - Select **Save**.

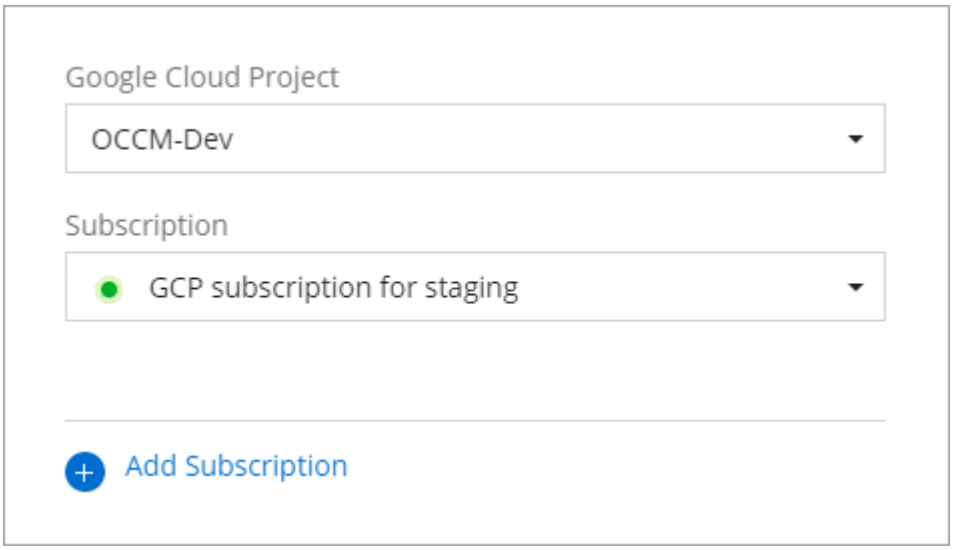
Google Cloud

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**.



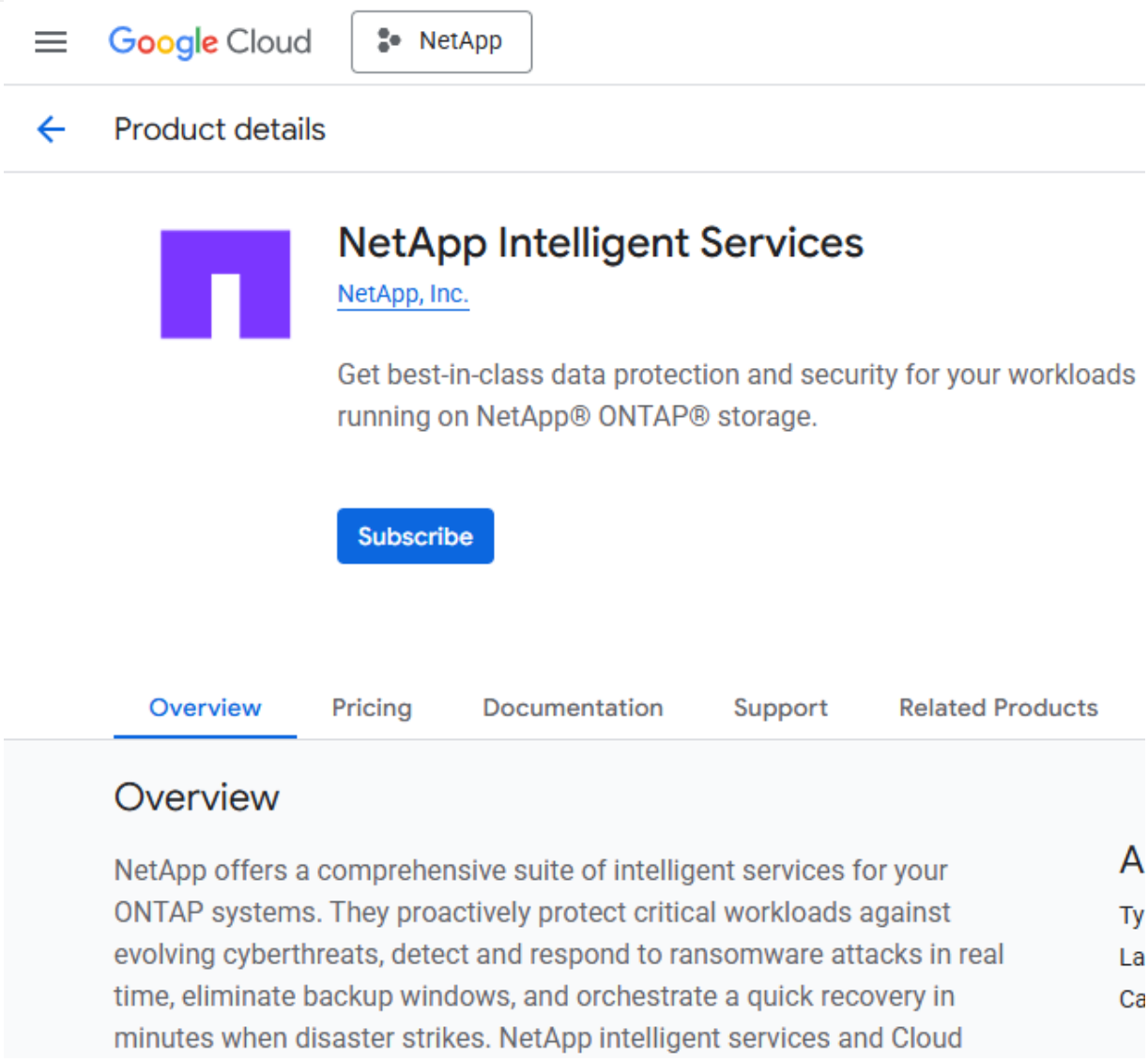
1. To configure an existing subscription with the selected credentials, select a Google Cloud project and subscription from the drop-down list, and then select **Configure**.



2. If you don’t already have a subscription, select **Add Subscription > Continue** and follow the steps in the Google Cloud Marketplace.

   Before you complete the following steps, ensure that you have both Billing Admin privileges in your Google Cloud account as well as a NetApp Console login.

   a. After you’re redirected to the NetApp Intelligent Services page on the Google Cloud Marketplace, ensure that the correct project is selected at the top navigation menu.

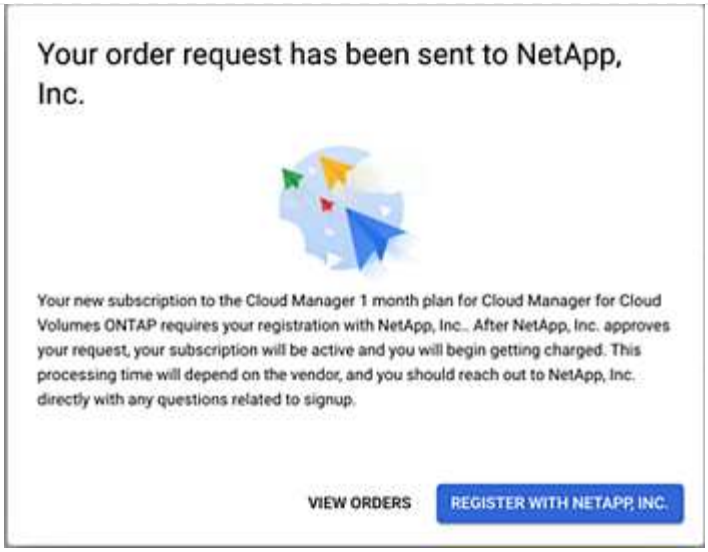


b. Select **Subscribe**.
c. Select the appropriate billing account and agree to the terms and conditions.
d. Select **Subscribe**.

   This step sends your transfer request to NetApp.

e. On the pop-up dialog box, select **Register with NetApp, Inc.**

   This step must be completed to link the Google Cloud subscription with your Console organization or account. The process of linking a subscription isn't complete until you are redirected from this page and then sign in to the Console.



f. Complete the steps on the **Subscription Assignment** page:

> If someone from your organization has already has a marketplace subscription from your billing account, then you will be redirected to the Cloud Volumes ONTAP page within NetApp Console instead. If this is unexpected, contact your NetApp sales team. Google enables only one subscription per Google billing account.

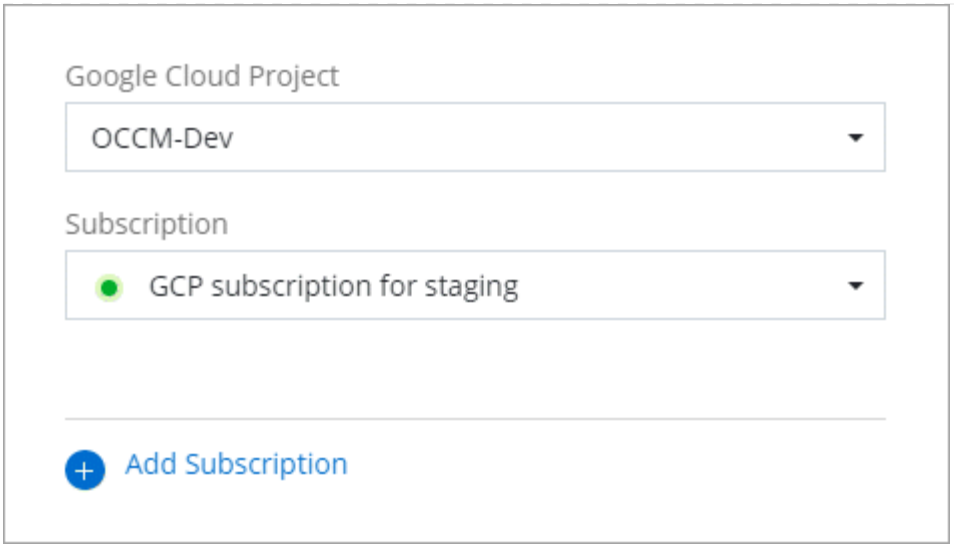
- Select the Console organization that you'd like to associate this subscription with.
- In the **Replace existing subscription** field, choose whether you'd like to automatically replace the existing subscription for one organization with this new subscription.

  The Console replaces the existing subscription for all credentials in the organization 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**.

a. Once this process is complete, navigate back to the Credentials page in the Console and select this new subscription.



Related information

- Manage BYOL capacity-based licenses for Cloud Volumes ONTAP
- Manage BYOL licenses for data services
- Manage AWS credentials and subscriptions
- Manage Azure credentials and subscriptions
- Manage Google Cloud credentials and subscriptions

What you can do next (restricted mode)

After you get up and running with NetApp Console in restricted mode, you can start using the services that are supported with restricted mode.

For help, refer to the documentation for these services:

- Azure NetApp Files docs
- Backup and recovery docs
- Classification docs
- Cloud Volumes ONTAP docs
- Digital wallet docs
- On-premises ONTAP cluster docs
- Replication docs

Related information

NetApp Console deployment modes

Copyright information

Trademark information