Manage data for Kubernetes clusters

Cloud Manager

NetApp
January 21, 2022

Table of Contents

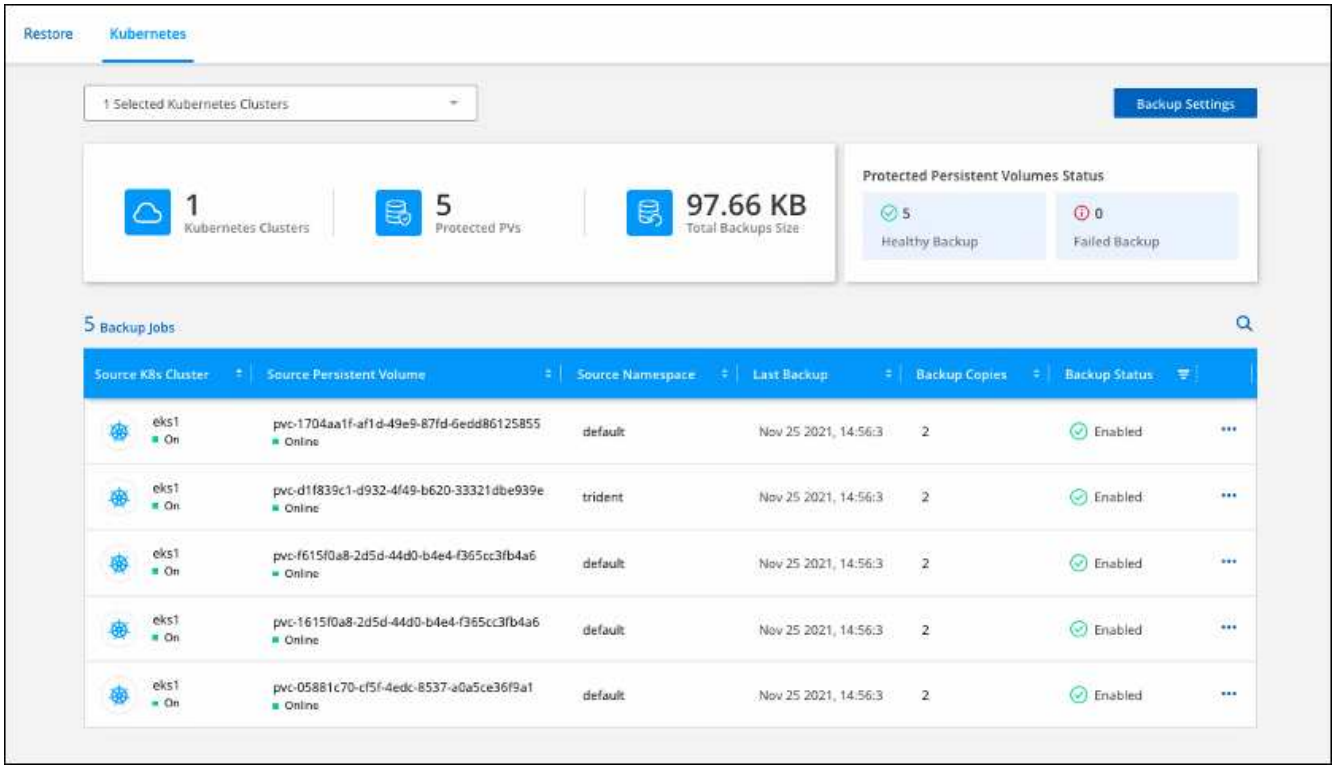
Manage data for Kubernetes clusters

Kubernetes overview

Add managed-Kubernetes clusters to Cloud Manager for advanced data management.

Features

- Add clusters to the Canvas to view and manage them as part of your hybrid cloud infrastructure
- Back up persistent volumes using Cloud Backup Service



Supported Kubernetes deployments

Cloud Manager supports managed-Kubernetes clusters running in Amazon Elastic Kubernetes Service (Amazon EKS) and Microsoft Azure Kubernetes Service (AKS).

Supported backend storage

NetApp's Astra Trident must be installed on each Kubernetes cluster and Cloud Volumes ONTAP must be configured as backend storage for the clusters.

Cost

There are no charges to *discover* your Kubernetes clusters in Cloud Manager, but you will be charged when you back up persistent volumes using Cloud Backup Service.

Get started with Amazon EKS clusters

Requirements for EKS clusters

Before you can add an Amazon Elastic Kubernetes Service (Amazon EKS) cluster to Cloud Manager, you need to ensure that the following requirements have been met.

Requirements

Astra Trident

The EKS cluster must have NetApp Astra Trident installed. One of the four most recent versions of Astra Trident is required. Go to the Astra Trident docs for installation steps.

Cloud Volumes ONTAP

Cloud Volumes ONTAP for AWS must be set up as backend storage for the cluster. Go to the Astra Trident docs for configuration steps.

Cloud Manager Connector

A Connector must be running in AWS with the required permissions. Learn more below.

Network connectivity

Network connectivity is required between the EKS cluster and the Connector and between the EKS cluster and Cloud Volumes ONTAP. Learn more below.

RBAC authorization

The Cloud Manager Connector role must be authorized on each EKS cluster. Learn more below.

Prepare a Connector

A Cloud Manager Connector is required in AWS to discover and manage Amazon EKS clusters. You'll need to create a new Connector or use an existing Connector that has the required permissions.

Create a new Connector

Follow the steps in one of the links below.

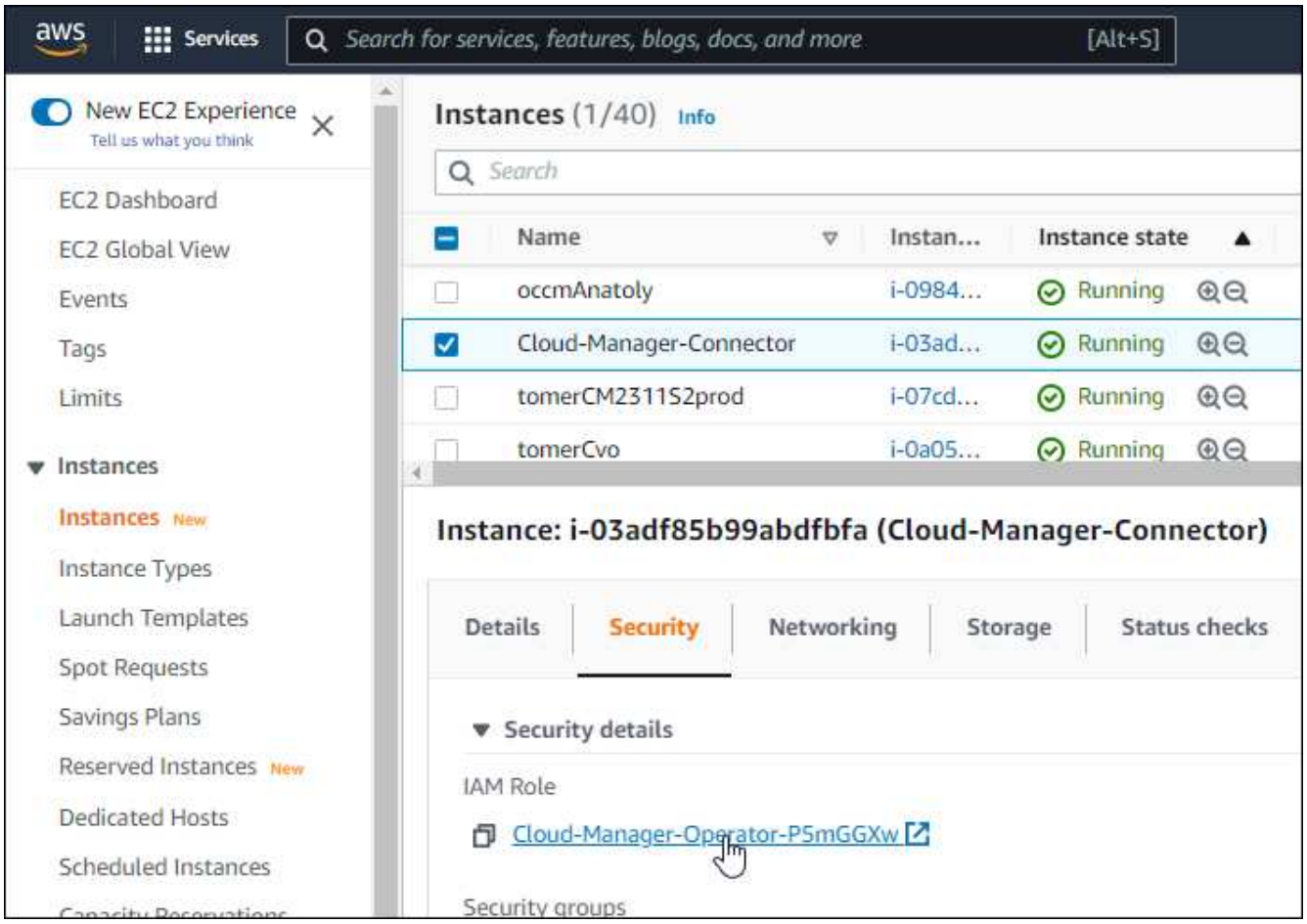
- Create a Connector from Cloud Manager (recommended)
- Create a Connector from the AWS Marketplace
- Install the Connector on an existing Linux host in AWS

Add the required permissions to an existing Connector

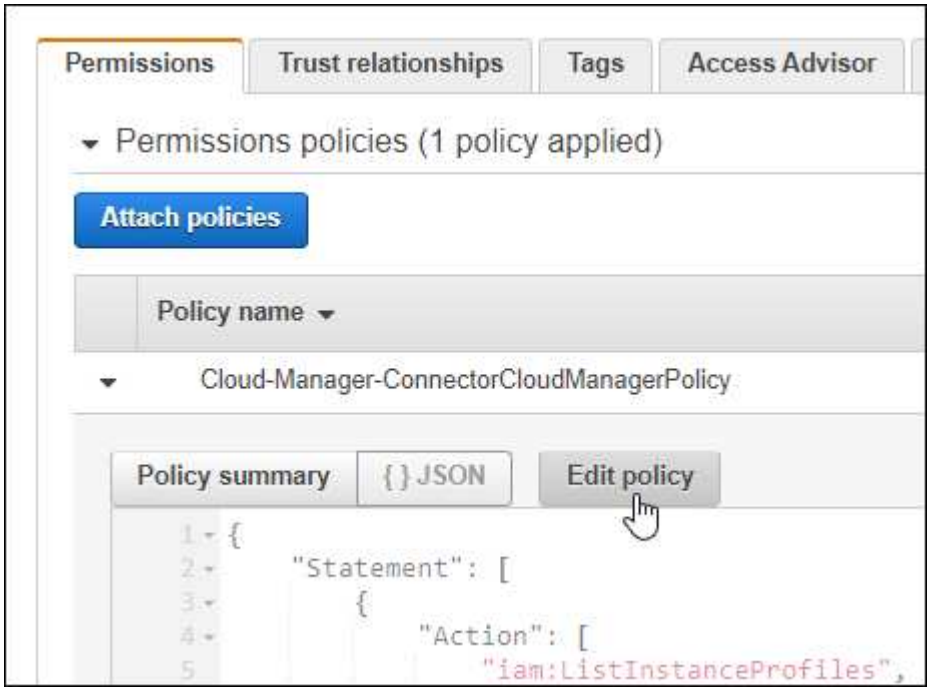
Starting in the 3.9.13 release, any *newly* created Connectors include three new AWS permissions that enable discovery and management of EKS clusters. If you created a Connector prior to this release, then you'll need to modify the existing policy for the Connector's IAM role to provide the permissions.

Steps

1. Go the AWS console and open the EC2 service.
2. Select the Connector instance, click **Security**, and click the name of the IAM role to view the role in the IAM service.



3. In the **Permissions** tab, expand the policy and click **Edit policy**.



4. Click **JSON** and add the following permissions under the first set of actions:

```
"eks:ListClusters",
"eks:DescribeCluster,"
"iam:GetInstanceProfile"
```

View the full JSON format for the policy.

5. Click **Review policy** and then click **Save changes**.

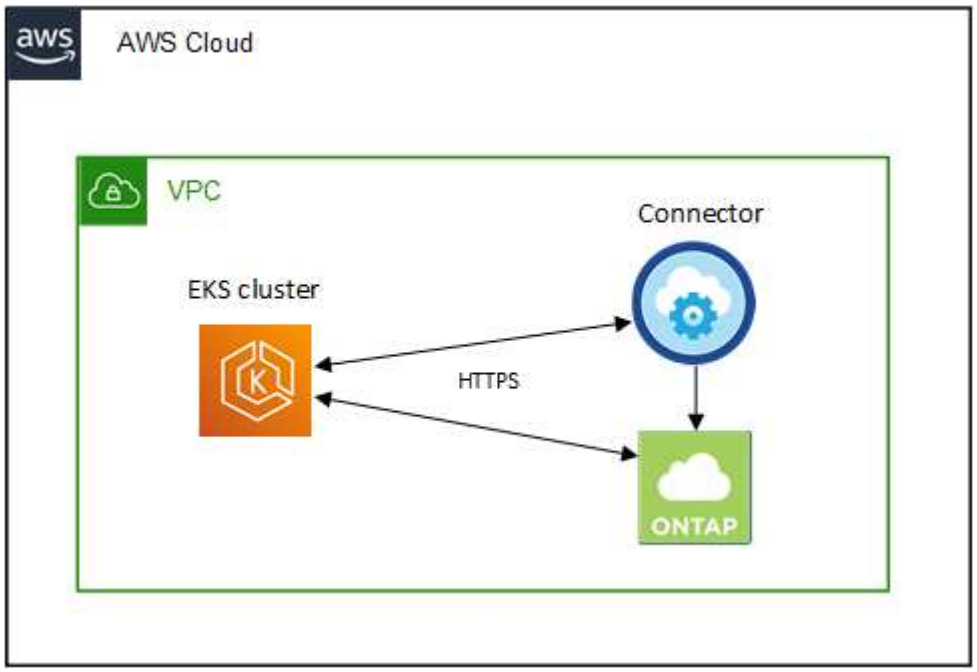
Review networking requirements

You need to provide network connectivity between the EKS cluster and the Connector and between the EKS cluster and the Cloud Volumes ONTAP system that provides backend storage to the cluster.

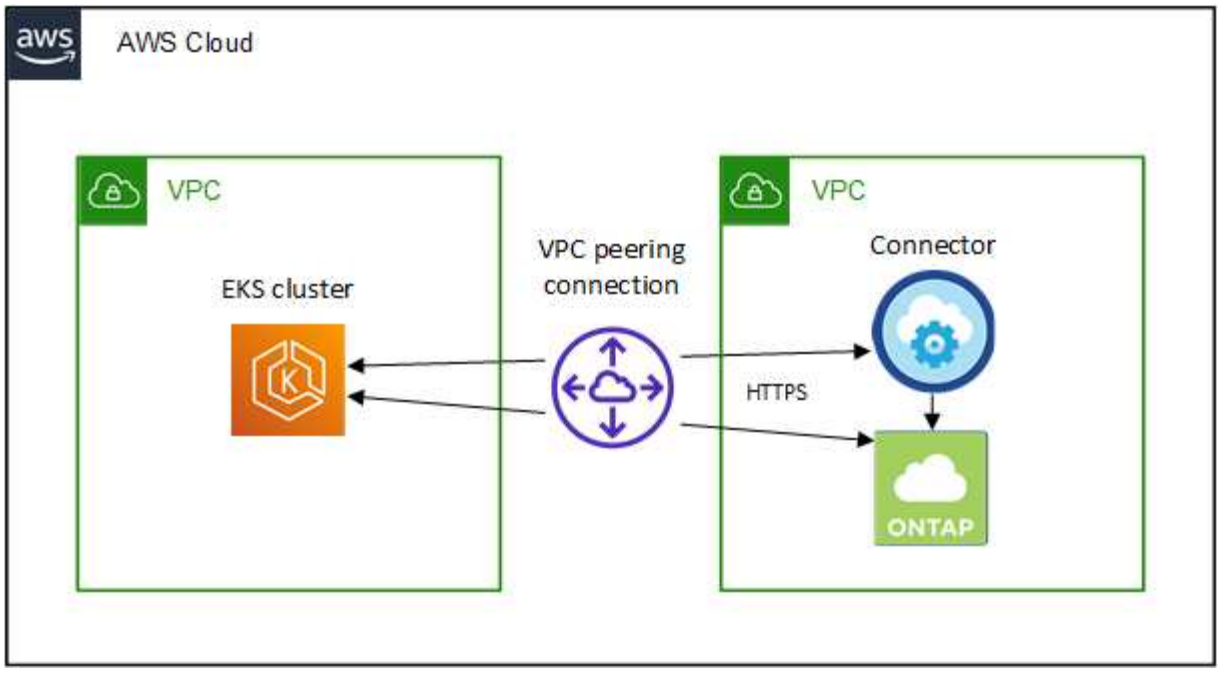
- Each EKS cluster must have an inbound connection from the Connector
- The Connector must have an outbound connection to eks.amazonaws.com on port 443

The simplest way to provide this connectivity is to deploy the Connector and Cloud Volumes ONTAP in the same VPC as the EKS cluster. Otherwise, you need to set up a VPC peering connection between the different VPCs.

Here's an example that shows each component in the same VPC.



And here's another example that shows an EKS cluster running in a different VPC. In this example, VPC peering provides a connection between the VPC for the EKS cluster and the VPC for the Connector and Cloud Volumes ONTAP.



Set up RBAC authorization

You need to authorize the Connector role on each EKS cluster so the Connector can discover and manage a cluster.

Steps

1. Create a cluster role and role binding.
    a. Create a YAML file that includes the following text.

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
    name: cloudmanager-access-clusterrole
rules:
    - apiGroups:
        - ''
      resources:
        - secrets
        - namespaces
        - persistentvolumeclaims
        - persistentvolumes
      verbs:
        - get
        - list
        - create
    - apiGroups:
        - storage.k8s.io
      resources:
        - storageclasses
      verbs:
        - get
        - list
    - apiGroups:
        - trident.netapp.io
      resources:
        - tridentbackends
        - tridentorchestrators
      verbs:
        - get
        - list
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
    name: k8s-access-binding
subjects:
    - kind: Group
      name: cloudmanager-access-group
      apiGroup: rbac.authorization.k8s.io
roleRef:
    kind: ClusterRole
    name: cloudmanager-access-clusterrole
    apiGroup: rbac.authorization.k8s.io
```

b. Apply the configuration to a cluster.

```
kubectl apply -f <file-name>
```

2. Create an identity mapping to the permissions group.

Use eksctl

Use eksctl to create an IAM identity mapping between a cluster and the IAM role for the Cloud Manager Connector.

Go to the eksctl documentation for full instructions.

An example is provided below.

```
eksctl create iamidentitymapping --cluster <eksCluster> --region
<us-east-2> --arn <ARN of the Connector IAM role> --group
cloudmanager-access-group --username
system:node:{{EC2PrivateDNSName}}
```

Edit aws-auth

Directly edit the aws-auth ConfigMap to add RBAC access to the IAM role for the Cloud Manager Connector.

Go to the Amazon EKS documentation for full instructions.

An example is provided below.

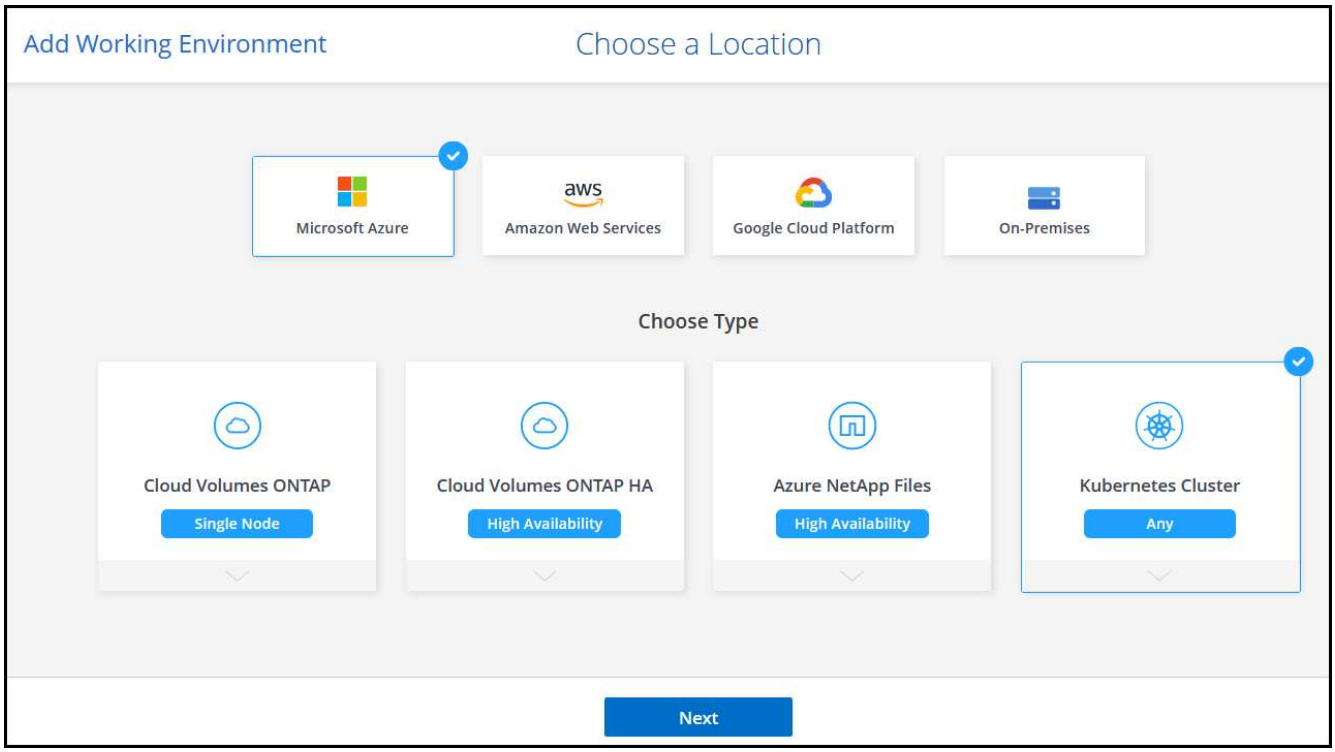
```
apiVersion: v1
data:
  mapRoles: |
    - groups:
      - cloudmanager-access-group
      rolearn: <ARN of the Connector IAM role>
     username: system:node:{{EC2PrivateDNSName}}
kind: ConfigMap
metadata:
  creationTimestamp: "2021-09-30T21:09:18Z"
  name: aws-auth
  namespace: kube-system
  resourceVersion: "1021"
  selfLink: /api/v1/namespaces/kube-system/configmaps/aws-auth
  uid: dcc31de5-3838-11e8-af26-02e00430057c
```

Add an Amazon EKS cluster to Cloud Manager

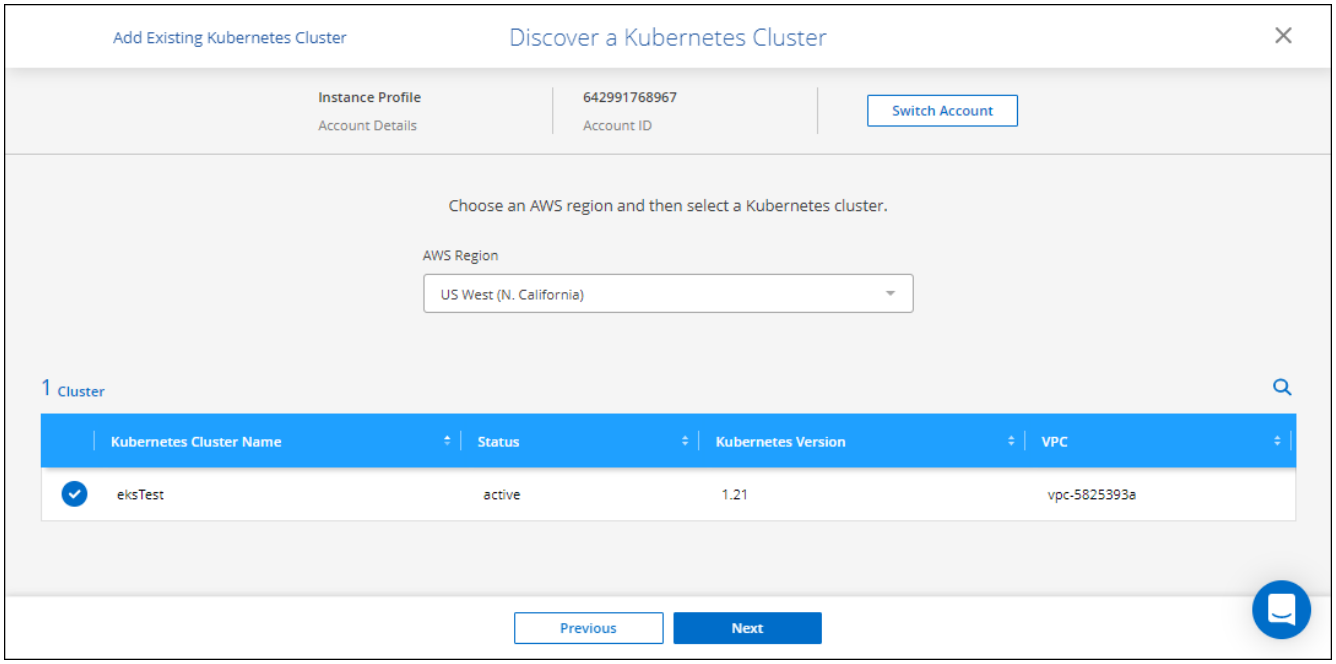
Add an Amazon Elastic Kubernetes Service (Amazon EKS) cluster to Cloud Manager so that you can start backing up persistent volumes to Amazon S3.

Steps

1. On the **Canvas**, click **Add Working Environment**.
2. Select **Amazon Web Services** > **Kubernetes Cluster** and click **Next**.

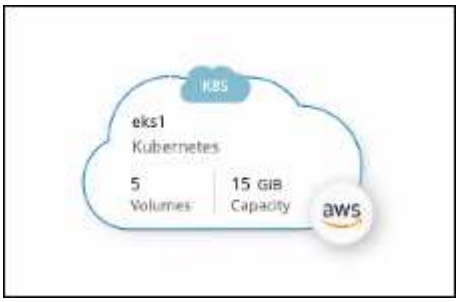


3. Select **Discover Cluster** and click **Next**.
4. Choose an AWS region, select a Kubernetes cluster, and then click **Next**.



Result

Cloud Manager adds the Kubernetes cluster to the Canvas.



Get started with Kubernetes clusters in Azure

Requirements for Kubernetes clusters in Azure

You can add and manage managed Azure Kubernetes clusters (AKS) and self-managed Kubernetes clusters in Azure using Cloud Manager. Before you can add the clusters to Cloud Manager, ensure the following requirements are met.

This topic uses *Kubernetes cluster* where configuration is the same for AKS and self-managed Kubernetes clusters. The cluster type is specified where configuration differs.

Requirements

Astra Trident

The Kubernetes cluster must have NetApp Astra Trident deployed. Install one of the four most recent versions of Astra Trident using Helm. Go to the Astra Trident docs for installation steps using Helm.

Cloud Volumes ONTAP

Cloud Volumes ONTAP must be set up as backend storage for the cluster. Go to the Astra Trident docs for configuration steps.

Cloud Manager Connector

A Connector must be running in Azure with the required permissions. Learn more below.

Network connectivity

Network connectivity is required between the Kubernetes cluster and the Connector and between the Kubernetes cluster and Cloud Volumes ONTAP. Learn more below.

RBAC authorization

Cloud Manager supports RBAC-enabled clusters with and without Active Directory. The Cloud Manager Connector role must be authorized on each Azure cluster. Learn more below.

Prepare a Connector

A Cloud Manager Connector in Azure is required to discover and manage Kubernetes clusters. You'll need to create a new Connector or use an existing Connector that has the required permissions.

Create a new Connector

Follow the steps in one of the links below.

- Create a Connector from Cloud Manager (recommended)
- Create a Connector from the Azure Marketplace
- Install the Connector on an existing Linux host

Add the required permissions to an existing Connector (to discover a managed AKS cluster)

If you want to discover a managed AKS cluster, you might need to modify the custom role for the Connector to provide the permissions.

Steps

1. Identify the role assigned to the Connector virtual machine:
    a. In the Azure portal, open the Virtual machines service.
    b. Select the Connector virtual machine.
    c. Under Settings, select **Identity**.
    d. Click **Azure role assignments**.
    e. Make note of the custom role assigned to the Connector virtual machine.
2. Update the custom role:
    a. In the Azure portal, open your Azure subscription.
    b. Click **Access control (IAM) > Roles**.
    c. Click the ellipsis (…) for the custom role and then click **Edit**.
    d. Click JSON and add the following permissions:

```
"Microsoft.ContainerService/managedClusters/listClusterUserCredential
/action"
"Microsoft.ContainerService/managedClusters/read"
```

e. Click **Review + update** and then click **Update**.

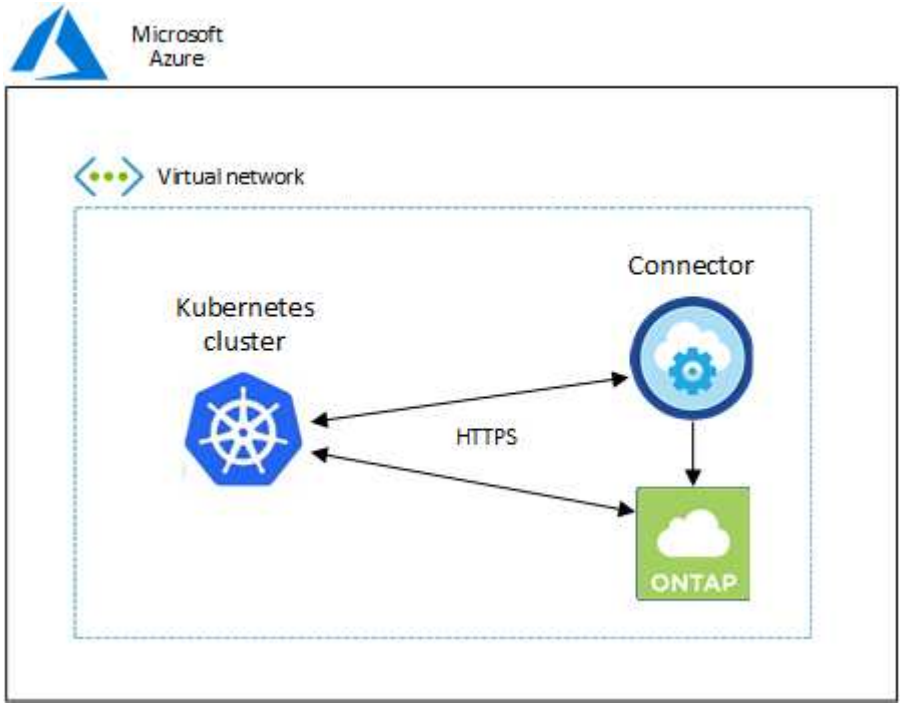
Review networking requirements

You need to provide network connectivity between the Kubernetes cluster and the Connector and between the Kubernetes cluster and the Cloud Volumes ONTAP system that provides backend storage to the cluster.

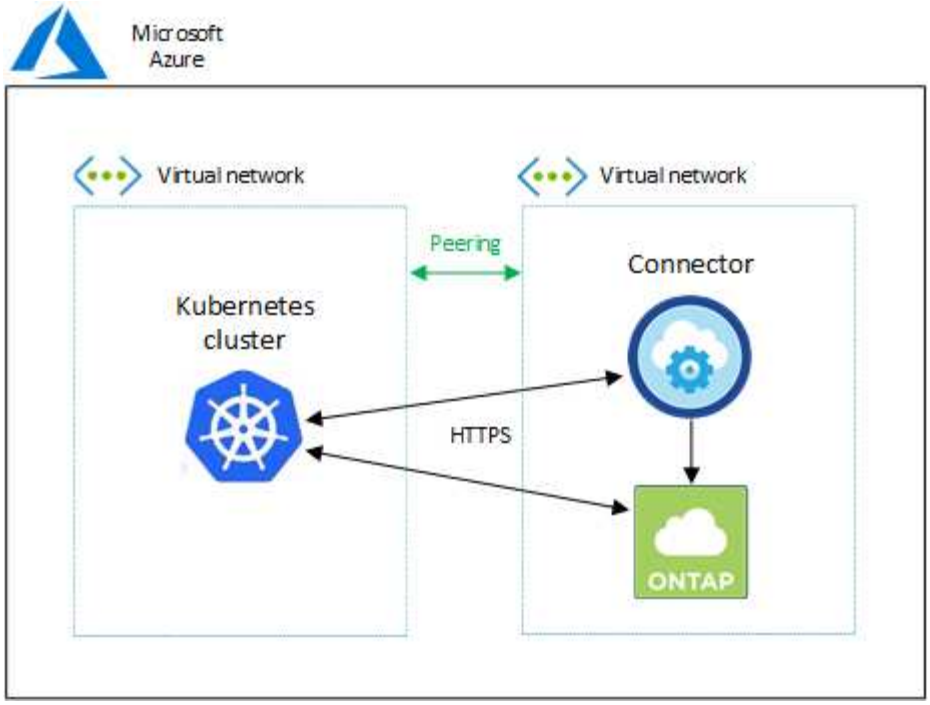
- Each Kubernetes cluster must have an inbound connection from the Connector
- The Connector must have an outbound connection to Kubernetes cluster over port 443

The simplest way to provide this connectivity is to deploy the Connector and Cloud Volumes ONTAP in the same VNet as the Kubernetes cluster. Otherwise, you need to set up a peering connection between the different VNets.

Here's an example that shows each component in the same VNet.



And here's another example that shows a Kubernetes cluster running in a different VNet. In this example, peering provides a connection between the VNet for the Kubernetes cluster and the VNet for the Connector and Cloud Volumes ONTAP.



Set up RBAC authorization

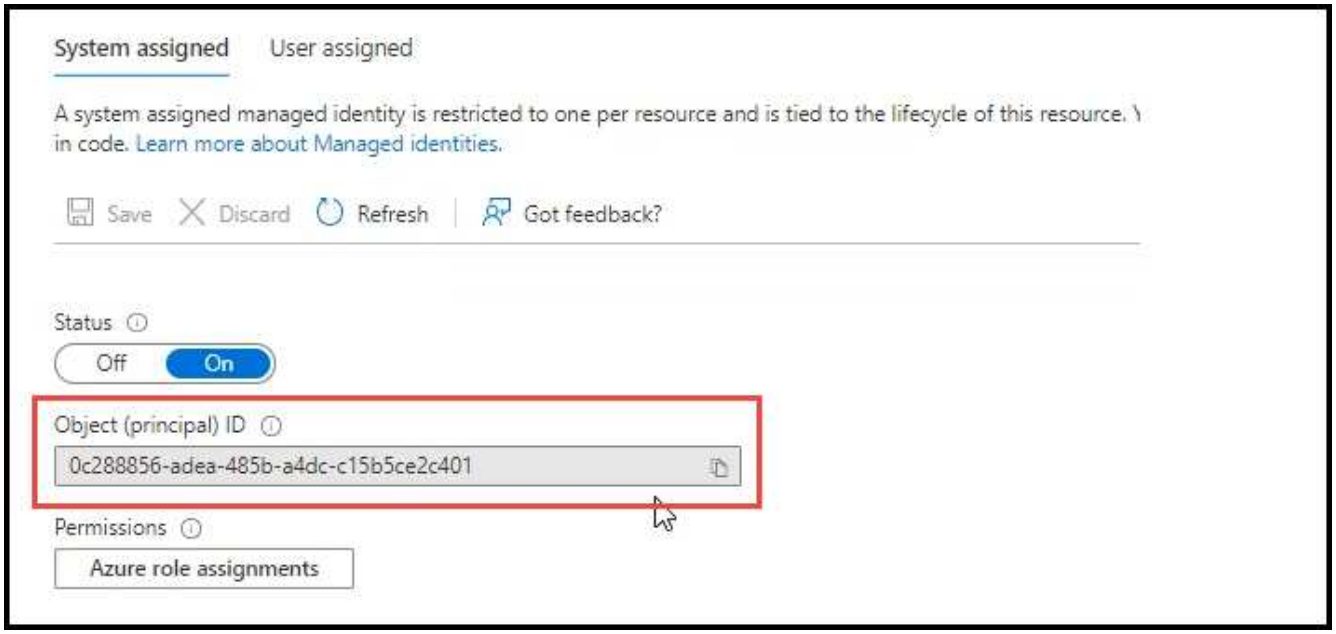
RBAC validation occurs only on Kubernetes clusters with Active Directory (AD) enabled. Kubernetes clusters without AD will pass validation automatically.

You need authorize the Connector role on each Kubernetes cluster so the Connector can discover and manage a cluster.

Before you begin

Your RBAC `subjects: name:` configuration varies slightly based on your Kubernetes cluster type.

- If you are deploying a **managed AKS cluster**, you need the Object ID for the system-assigned managed identity for the Connector. This ID is available in Azure management portal.



- If you are deploying a **self-managed Kubernetes cluster**, you need the username of any authorized user.

Steps

1. Create a cluster role and role binding.

    a. Create a YAML file that includes the following text. Replace the `subjects: kind:` variable with your username and `subjects: user:` with either the Object ID for the system-assigned managed identity or username of any authorized user as described above.

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
    name: cloudmanager-access-clusterrole
rules:
    - apiGroups:
        - ''
      resources:
        - secrets
        - namespaces
        - persistentvolumeclaims
        - persistentvolumes
      verbs:
        - get
        - list
        - create
    - apiGroups:
        - storage.k8s.io
      resources:
        - storageclasses
      verbs:
        - get
        - list
    - apiGroups:
        - trident.netapp.io
      resources:
        - tridentbackends
        - tridentorchestrators
      verbs:
        - get
        - list
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
    name: k8s-access-binding
subjects:
    - kind: User
      name: Object (principal) ID (for AKS) or username (for self-
managed)
      apiGroup: rbac.authorization.k8s.io
roleRef:
    kind: ClusterRole
    name: cloudmanager-access-clusterrole
    apiGroup: rbac.authorization.k8s.io
```

b. Apply the configuration to a cluster.

```
kubectl apply -f <file-name>
```

Add an Azure Kubernetes cluster to Cloud Manager

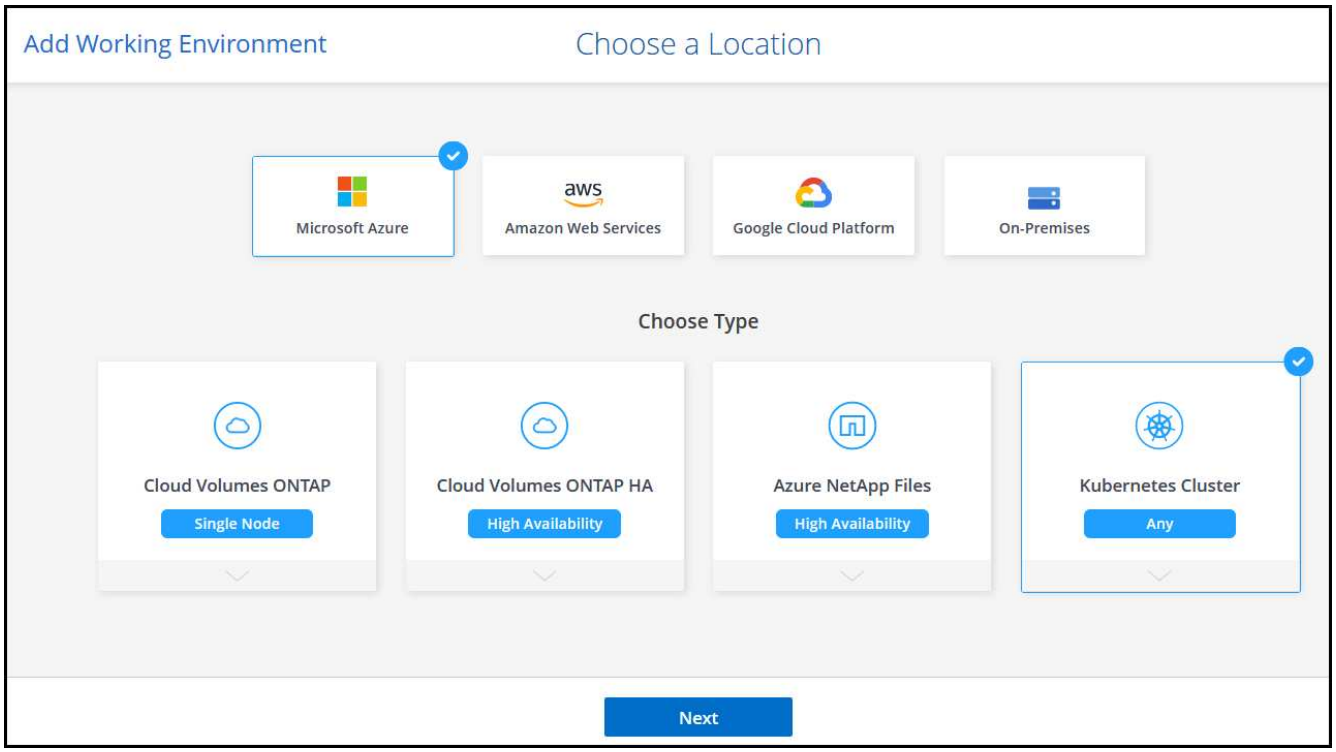
You can discover or import Kubernetes clusters to Cloud Manager so that you can start backing up persistent volumes to Azure.

Discover a cluster

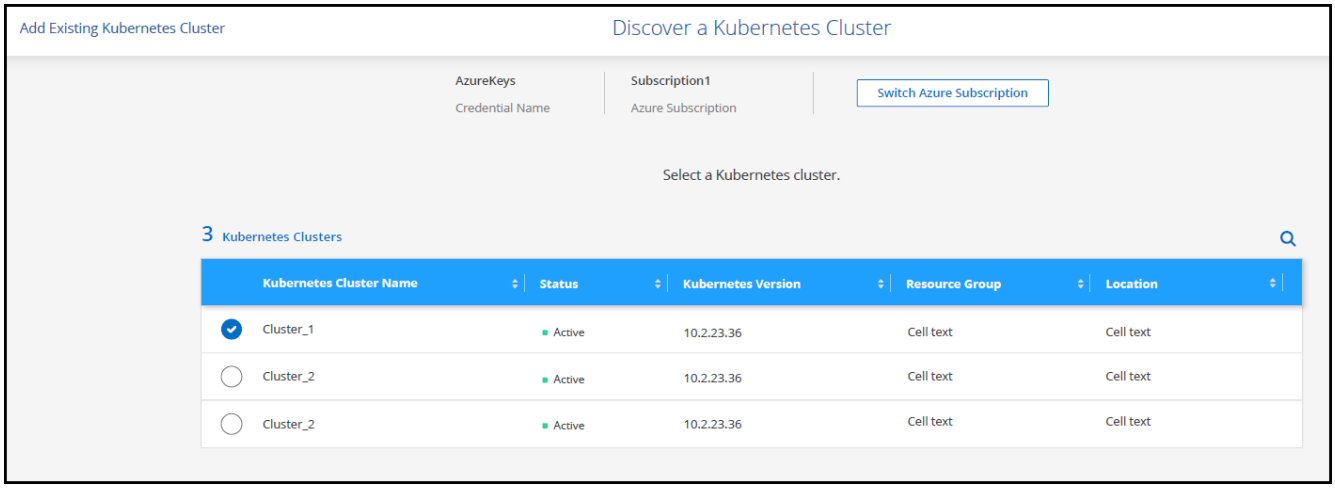
You can discover a fully-managed or self-managed Kubernetes cluster. Managed clusters must be discovered; they cannot be imported.

Steps

1. On the **Canvas**, click **Add Working Environment**.
2. Select **Microsoft Azure** > **Kubernetes Cluster** and click **Next**.

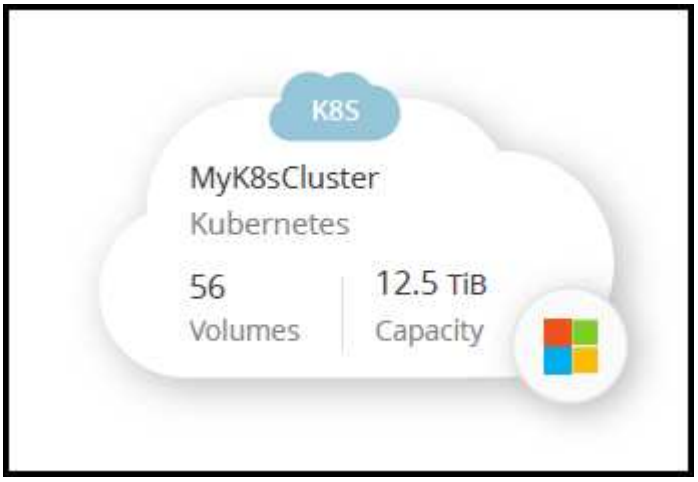


3. Select **Discover Cluster** and click **Next**.
4. Select a Kubernetes cluster and click **Next**.



Result

Cloud Manager adds the Kubernetes cluster to the Canvas.



Import a Cluster

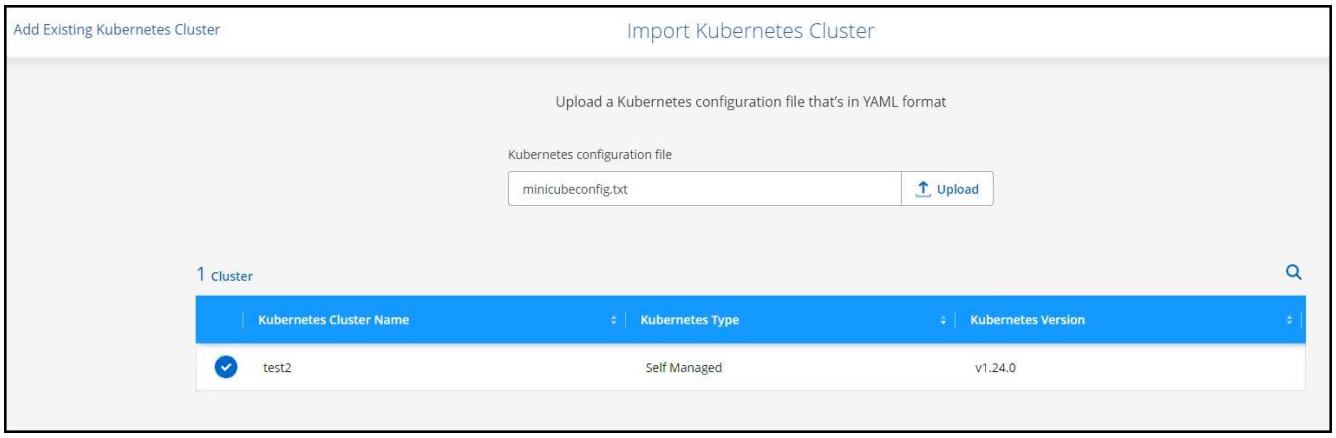
You can import a self-managed Kubernetes cluster using a Kubernetes configuration file.

Before you get started

You will need Certificate Authority, Client Key, and Client Certificate certificates for the user specified in the cluster role YAML file to import Kubernetes clusters. The Kubernetes cluster administrator receives these certifications when creating users on the Kubernetes cluster.

Steps

1. On the **Canvas**, click **Add Working Environment**.
2. Select **Microsoft Azure** > **Kubernetes Cluster** and click **Next**.
3. Select **Import Cluster** and click **Next**.
4. Upload a Kubernetes configuration file in YAML format.



5. Upload the cluster certificates provided by your Kubernetes cluster administrator.

Result

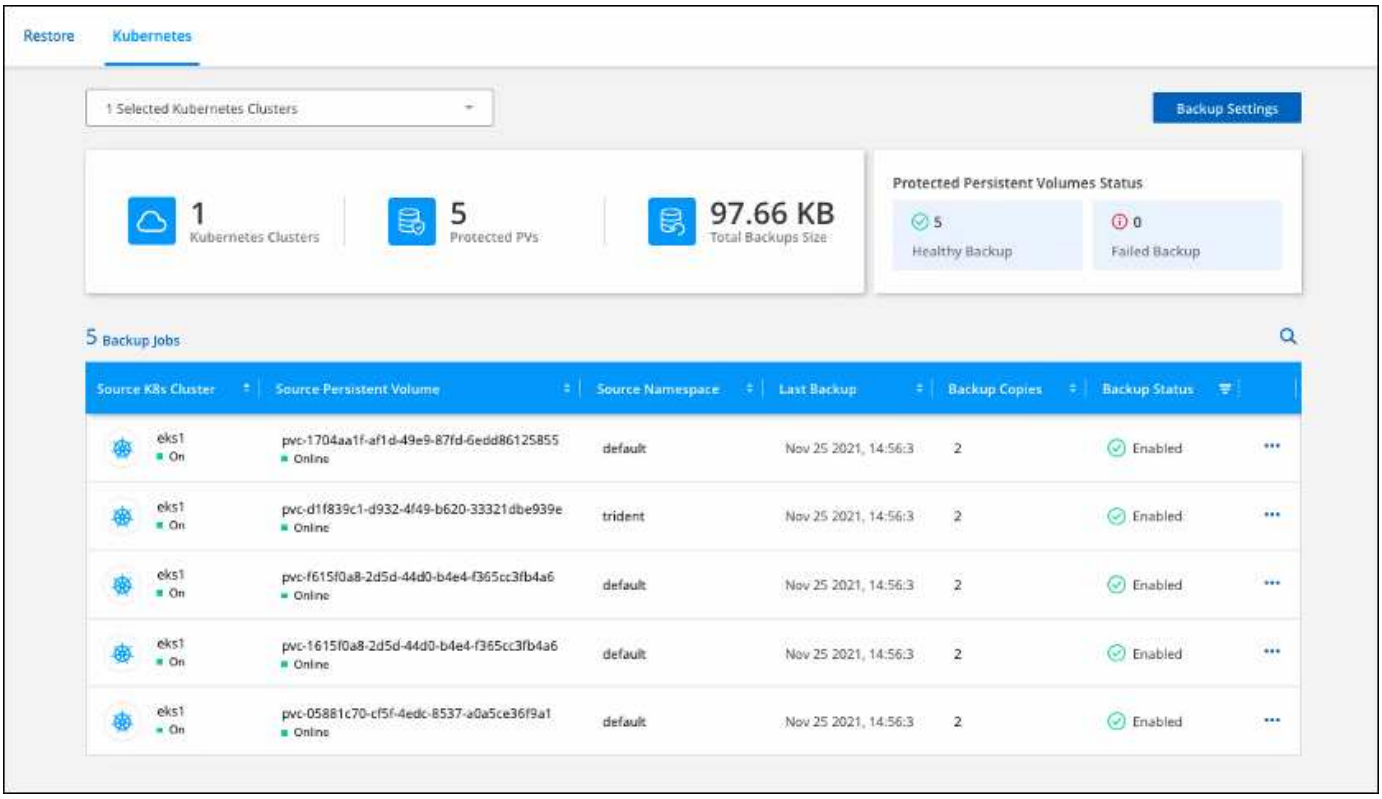
Cloud Manager adds the Kubernetes cluster to the Canvas.

Use NetApp's cloud data services with Kubernetes clusters

After you add a managed-Kubernetes cluster to the Canvas, you can use NetApp's cloud data services for advanced data management.

At this time, Cloud Backup is supported with Kubernetes clusters. You can use Cloud Backup to back up persistent volumes to object storage.

Go to the Cloud Backup docs to learn how to back up persistent volumes.


Restore
Kubernetes
1 Selected Kubernetes Clusters
Backup Settings
1
Kubernetes Clusters
5
Protected PVs
97.66 KB
Total Backups Size
Protected Persistent Volumes Status
5
Healthy Backup
0
Failed Backup
5 Backup Jobs
Source K8s Cluster
Source Persistent Volume
Source Namespace
Last Backup
Backup Copies
Backup Status
eks1
On
pvc-1704aa1f-af1d-49e9-87fd-6edd86125855
Online
default
Nov 25 2021, 14:56:3
2
Enabled
eks1
On
pvc-d1f839c1-d932-4f49-b620-33321dbe939e
Online
trident
Nov 25 2021, 14:56:3
2
Enabled
eks1
On
pvc-f615f0a8-2d5d-44d0-b4e4-f365cc3fb4a6
Online
default
Nov 25 2021, 14:56:3
2
Enabled
eks1
On
pvc-1615f0a8-2d5d-44d0-b4e4-f365cc3fb4a6
Online
default
Nov 25 2021, 14:56:3
2
Enabled
eks1
On
pvc-05881c70-cf5f-4edc-8537-a0a5ce36f9a1
Online
default
Nov 25 2021, 14:56:3
2
Enabled