



# **NetApp Hybrid Multicloud with Red Hat OpenShift**

NetApp Solutions

NetApp  
September 19, 2024

This PDF was generated from <https://docs.netapp.com/us-en/netapp-solutions/rhhc/rhhc-value-prop.html> on September 19, 2024. Always check docs.netapp.com for the latest.

# Table of Contents

- NetApp Hybrid Multicloud with Red Hat OpenShift Container workloads ..... 1
  - NetApp Hybrid Multicloud solutions for Red Hat OpenShift Container workloads ..... 1
  - NetApp Hybrid Multicloud solutions for Red Hat OpenShift Container workloads ..... 13
  - NetApp Hybrid Multicloud solutions for Red Hat OpenShift Container workloads ..... 24
  - NetApp Hybrid Multicloud solutions for Red Hat OpenShift Container workloads ..... 41
- Data protection for Container Apps in OpenShift Container Platform using OpenShift API for Data Protection (OADP) ..... 65

# NetApp Hybrid Multicloud with Red Hat OpenShift Container workloads

## NetApp Hybrid Multicloud solutions for Red Hat OpenShift Container workloads

NetApp is seeing a significant increase in customers modernizing their legacy enterprise applications and building new applications using containers and orchestration platforms built around Kubernetes. Red Hat OpenShift Container Platform is one example that we see adopted by many of our customers.

### Overview

As more and more customers begin adopting containers within their enterprises, NetApp is perfectly positioned to help serve the persistent storage needs of their stateful applications and classic data management needs such as data protection, data security, and data migration. However, these needs are met using different strategies, tools, and methods.

**NetApp ONTAP** based storage options listed below, deliver security, data protection, reliability, and flexibility for containers and Kubernetes deployments.

- Self-managed storage in on-premises:
  - NetApp Fabric Attached Storage (FAS), NetApp All Flash FAS Arrays (AFF), NetApp All SAN Array (ASA) and ONTAP Select
- Provider-managed storage in on-premises:
  - NetApp Keystone provides Storage as a Service (STaaS)
- Self-managed storage in the cloud:
  - NetApp Cloud Volumes ONTAP(CVO) provide self managed storage in the hyperscalers
- Provider-managed storage in the cloud:
  - Cloud Volumes Service for Google Cloud (CVS), Azure NetApp Files (ANF), Amazon FSx for NetApp ONTAP offer fully managed storage in the hyperscalers

## ONTAP feature highlights



<b>Storage Administration</b> <ul style="list-style-type: none"><li>Multi-tenancy</li><li>FlexVol &amp; FlexGroup</li><li>LUN</li><li>Quotas</li><li>ONTAP CLI &amp; API</li><li>System Manager &amp; BlueXP</li></ul>	<b>Performance &amp; Scalability</b> <ul style="list-style-type: none"><li>FlexCache</li><li>FlexClone</li><li>nconnect, session trunking, multipathing</li><li>Scale-out clusters</li></ul>
<b>Availability &amp; Resilience</b> <ul style="list-style-type: none"><li>Multi-AZ HA deployment (MetroCluster)</li><li>SnapShot &amp; SnapRestore</li><li>SnapMirror</li><li>SnapMirror Business Continuity (MetroCluster)</li><li>SnapMirror Cloud</li></ul>	<b>Access Protocols</b> <ul style="list-style-type: none"><li>NFS –v3, v4, v4.1, v4.2</li><li>SMB – v2, v3</li><li>iSCSI</li><li>Multi-protocol access</li></ul>
<b>Storage Efficiency</b> <ul style="list-style-type: none"><li>Deduplication &amp; Compression</li><li>Compaction</li><li>Thin provisioning</li><li>Data Tiering (Fabric Pool)</li></ul>	<b>Security &amp; Compliance</b> <ul style="list-style-type: none"><li>Fpolicy &amp; Vscan</li><li>Active Directory integration</li><li>LDAP &amp; Kerberos</li><li>Certificate based authentication</li></ul>

**NetApp BlueXP** enables you to manage all of your storage and data assets from a single control plane/interface.

You can use BlueXP to create and administer cloud storage (for example, Cloud Volumes ONTAP and Azure NetApp Files), to move, protect, and analyze data, and to control many on-prem and edge storage devices.

**NetApp Astra Trident** is a CSI Compliant Storage Orchestrator that enable quick and easy consumption of persistent storage backed by a variety of the above-mentioned NetApp storage options. It is an open-source software maintained and supported by NetApp.

## Astra Trident CSI feature highlights



<b>CSI specific</b> <ul style="list-style-type: none"><li>CSI NetApp® Snapshot™ copies and volume creation from CSI Snapshot copies</li><li>CSI topology</li><li>Volume expansion</li></ul>	<b>Security</b> <ul style="list-style-type: none"><li>Dynamic-export policy management</li><li>iSCSI initiator-groups dynamic management</li><li>iSCSI bidirectional CHAP</li></ul>
<b>Control</b> <ul style="list-style-type: none"><li>Storage and performance consumption</li><li>Monitoring</li><li>Volume Import</li><li>Cross Namespace Volume Access</li></ul>	<b>Installation methods</b> <ul style="list-style-type: none"><li>Binary</li><li>Helm chart</li><li>Operator</li><li>GitOps</li></ul>
<b>Choose your access mode</b> <ul style="list-style-type: none"><li>RWO (ReadWriteOnce, i.e 1↔1)</li><li>RWX (ReadWriteMany, i.e 1↔n)</li><li>ROX (ReadOnlyMany)</li><li>RWOP (ReadWriteOnce POD)</li></ul>	<b>Choose your protocol</b> <ul style="list-style-type: none"><li>NFS</li><li>SMB</li><li>iSCSI</li></ul>

Business critical container workloads need more than just persistent volumes. Their data management requirements require protection and migration of the application kubernetes objects as well.



Application data includes kubernetes objects in addition to the user data: Some examples are as follows:

- kubernetes objects such as pods specs, PVCs, deployments, services
- custom config objects such as config maps and secrets
- persistent data such as Snapshot copies, backups, clones
- custom resources such as CRs and CRDs

**NetApp Astra Control**, available as both fully-managed and self-managed software, provides orchestration for robust application data management. Refer to the [Astra documentation](#) for additional details on the Astra family of products.

This reference documentation provides validation of migration and protection of container-based applications, deployed on RedHat OpenShift container platform, using NetApp Astra Control Center. In addition, the solution provides high-level details for the deployment and the use of Red Hat Advanced Cluster Management (ACM) for managing the container platforms. The document also highlights the details for the integration of NetApp storage with Red Hat OpenShift container platforms using Astra Trident CSI provisioner. Astra Control Center is deployed on the hub cluster and is used to manage the container applications and their persistent storage lifecycle. Finally, it provides a solution for replication and failover and fail-back for container workloads on managed Red Hat OpenShift clusters in AWS (ROSA) using Amazon FSx for NetApp ONTAP (FSxN) as persistent storage.

## Value propositions of NetApp Hybrid Multicloud solutions for Red Hat OpenShift Container workloads

Most customers do not just start out building Kubernetes based environments without any existing infrastructure. Perhaps they are a traditional IT shop running most of their enterprise applications on virtual machines (in large VMware environments for example). Then they start building small container-based environments to satisfy the needs of their modern application development teams. These initiatives usually start small and begin to become more pervasive as the teams learn these new technologies and skills, and begin to recognize the many benefits of adopting them.

The good news for customers is that NetApp can serve the needs of both environments. This set of solutions for hybrid multicloud with Red Hat OpenShift will empower NetApp customers to adopt modern cloud technologies and services without having to overhaul their entire infrastructure and organization. Whether customer applications and data are hosted on-premises, in cloud, run on virtual machines, or on containers, NetApp can provide consistent data management, protection, security, and portability. With these new solutions, the same value NetApp has delivered in on-premises data center environments for decades will be available across the enterprise entire data horizon, without requiring significant investment to retool, acquire new skills, or build new teams. NetApp is positioned well to help customers solve these business challenges regardless of what phase of their cloud journey they are in.

NetApp Hybrid Multi-Cloud with Red Hat Openshift:

- Gives customers validated designs and practices which demonstrate the best ways for customers to manage, protect, secure, and migrate their data and applications when using Red Hat OpenShift with

NetApp based storage solutions.

- Present best practices for customers running Red Hat OpenShift with NetApp storage in VMware environments, bare metal infrastructure, or a combination of both.
- Demonstrate strategies and options for both on-prem and cloud environments, as well as hybrid environments where both are used.

## **Supported Solutions of NetApp Hybrid Multicloud for Red Hat OpenShift Container workloads**

The solution tests and validates Migration & Centralized Data Protection with OpenShift container platform (OCP), OpenShift Advanced Cluster Manager (ACM), NetApp ONTAP, NetApp BlueXP and NetApp Astra Control Center (ACC).

For this solution, the following scenarios are tested and validated by NetApp. The solution is separated into multiple scenarios based on the following characteristics:

- on-premises
- cloud
  - self-managed OpenShift clusters and self-managed NetApp storage
  - provider-managed OpenShift clusters and provider-managed NetApp storage

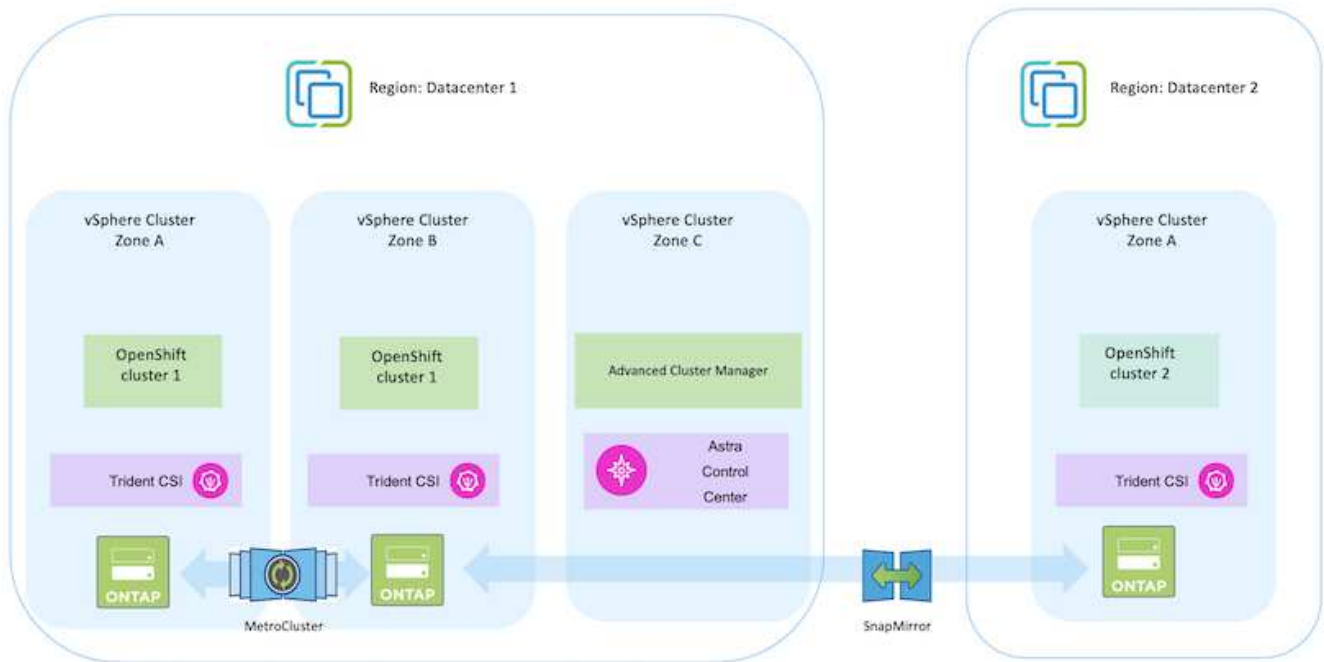
**We will be building out additional solutions and use cases in the future.**

### **Scenario 1: Data protection and migration within the on-premises environment using ACC**

#### **On-premises: self-managed OpenShift clusters and self-managed NetApp storage**

- Using ACC, create Snapshot copies, backups and restores for data protection.
- Using ACC, perform a SnapMirror replication of container applications.

#### **Scenario 1**



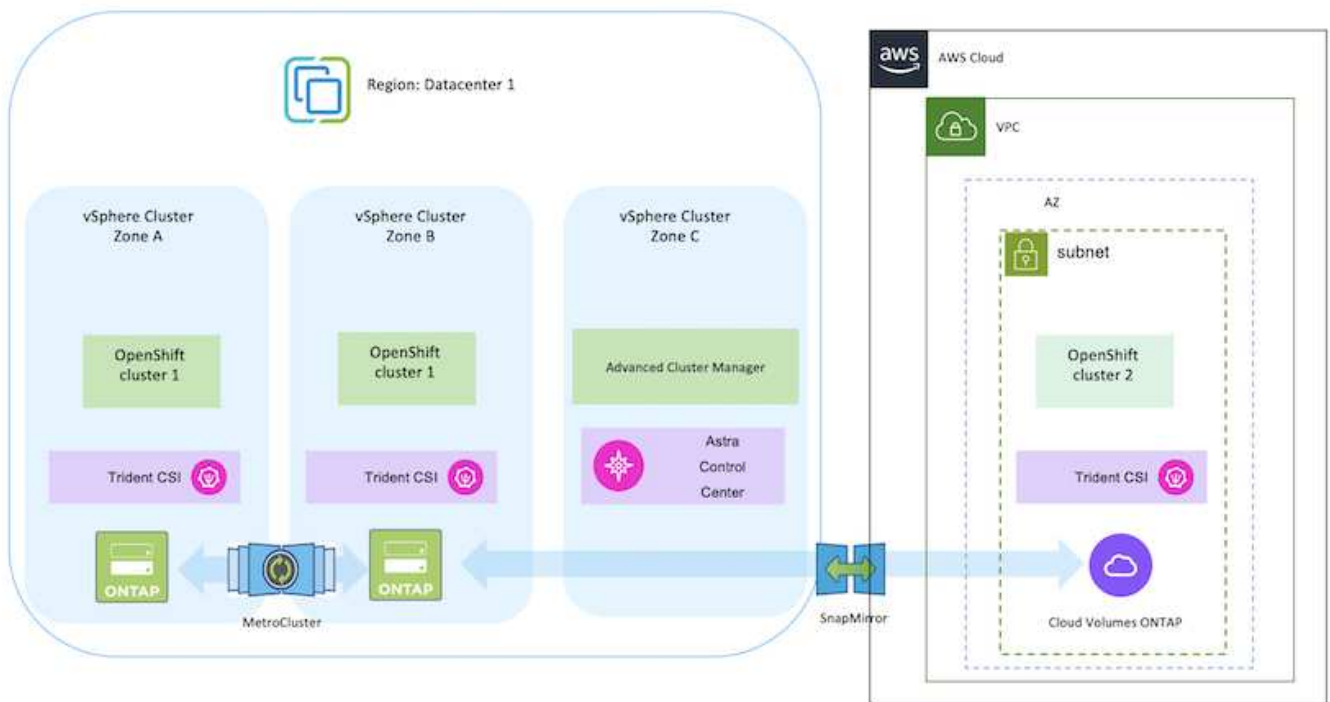
## Scenario 2: Data protection and migration from the on-premises environment to AWS environment using ACC

**On-premises: Self-managed OpenShift cluster and self-managed storage**

**AWS Cloud: Self-managed OpenShift cluster and self-managed storage**

- Using ACC, perform backups and restores for data protection.
- Using ACC, perform a SnapMirror replication of container applications.

### Scenario 2



### Scenario 3: Data protection and migration from the on-premises environment to AWS environment

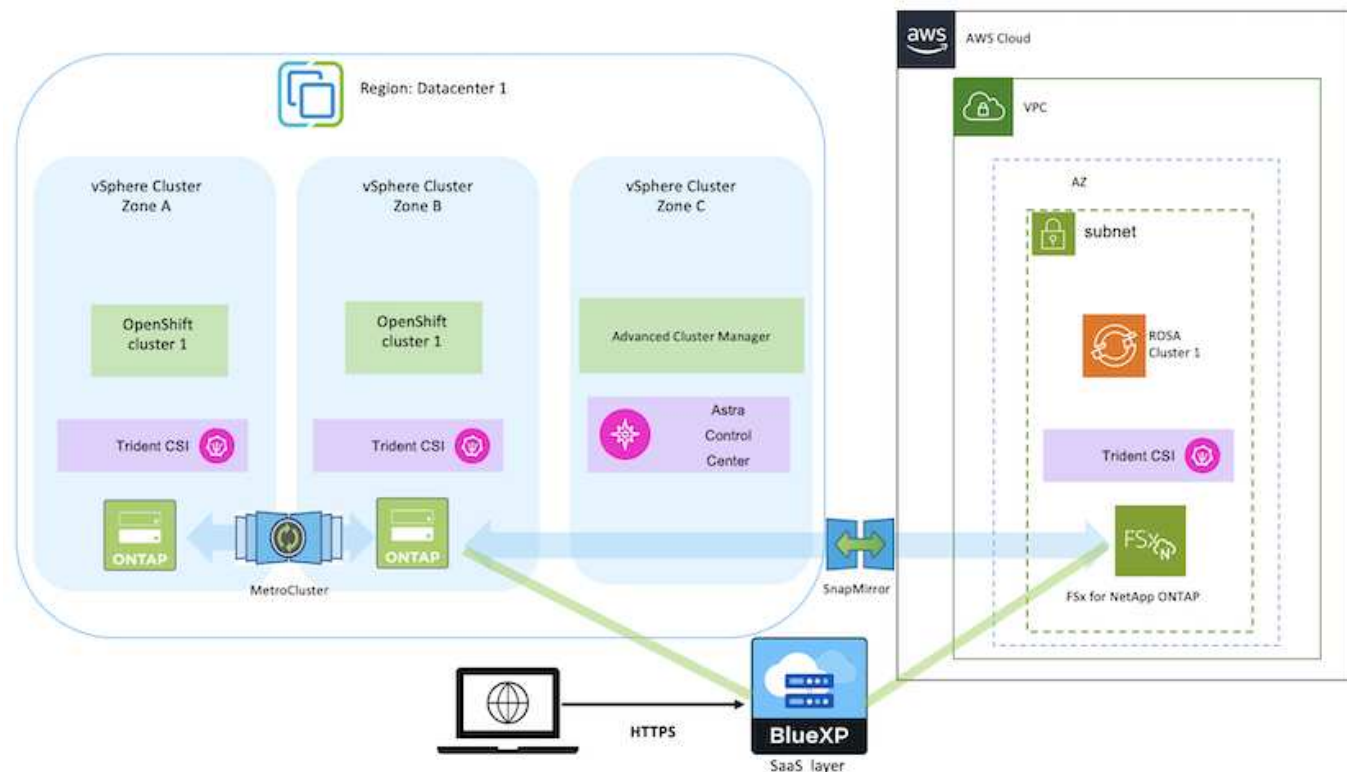
**On-premises: Self-managed OpenShift cluster and self-managed storage**

**AWS Cloud: Provider-managed OpenShift cluster (ROSA) and provider-managed storage (FSxN)**

- Using BlueXP, perform replication of persistent volumes (FSxN).
- Using OpenShift GitOps, recreate application metadata.

### Scenario 3



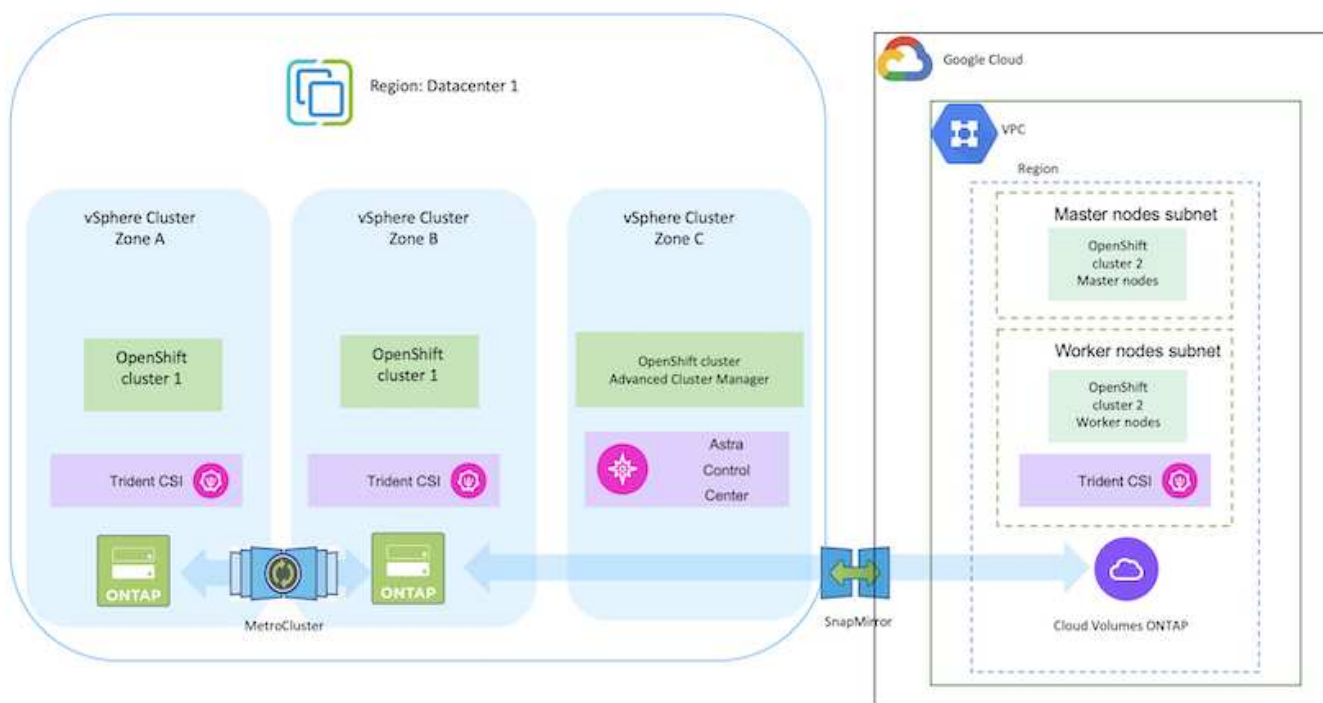


#### Scenario 4: Data protection and migration from the on-premises environment to GCP environment using ACC

**On-premises: Self-managed OpenShift cluster and self-managed storage**

**Google Cloud: Self-managed OpenShift cluster and self-managed storage**

- Using ACC, perform backups and restores for data protection.
- Using ACC, perform a SnapMirror replication of container applications.

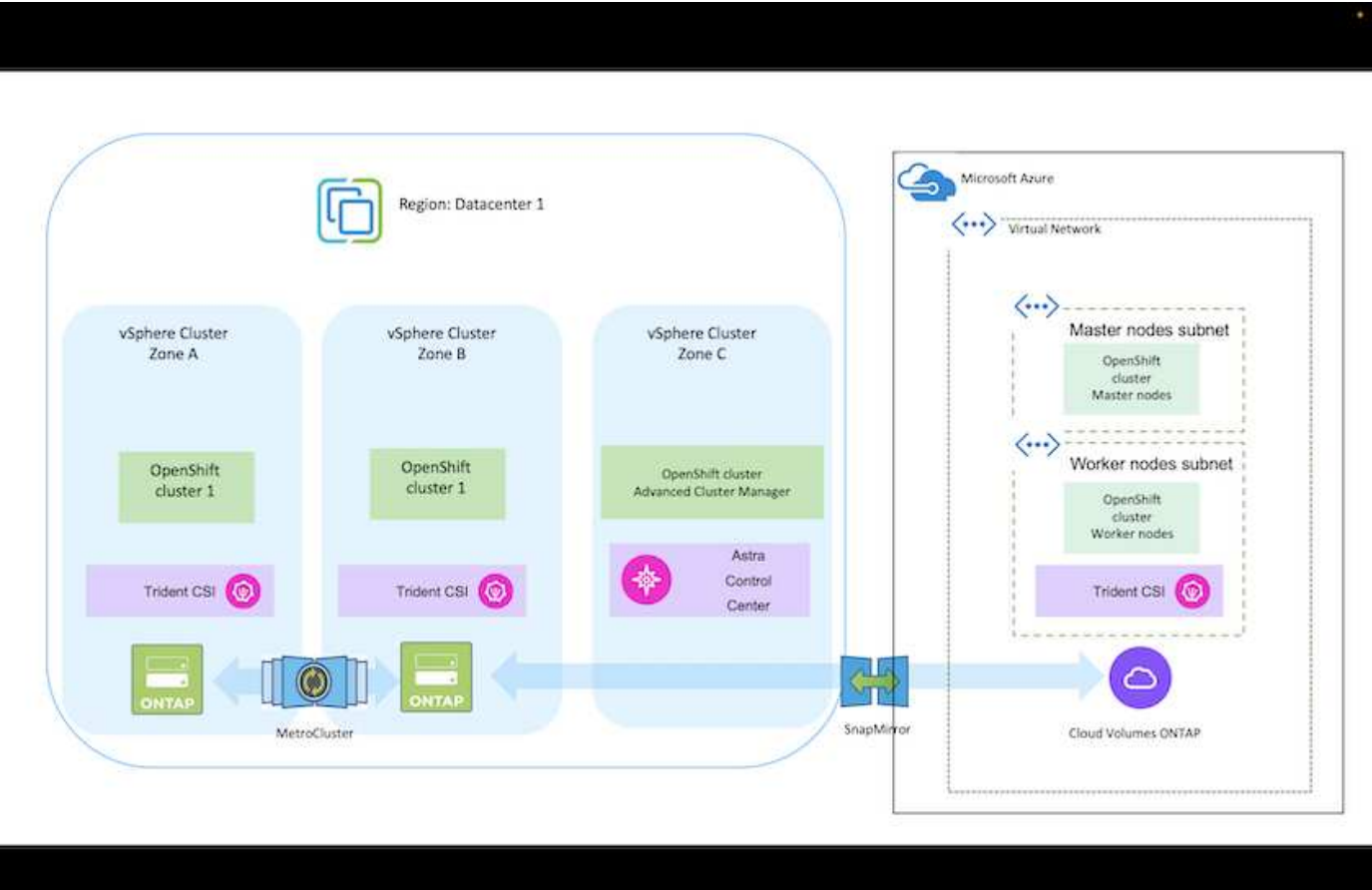


For considerations when using ONTAP in a MetroCluster configuration, refer [here](#).

**Scenario 5: Data protection and migration from the on-premises environment to Azure environment using ACC**

**On-premises: Self-managed OpenShift cluster and self-managed storage**  
**Azure Cloud: Self-managed OpenShift cluster and self-managed storage**

- Using ACC, perform backups and restores for data protection.
- Using ACC, perform a SnapMirror replication of container applications.



For considerations when using ONTAP in a MetroCluster configuration, refer [here](#).

**Versions of various components used in the solution validation**

The solution tests and validates Migration & Centralized Data Protection with OpenShift container platform, OpenShift Advanced Cluster Manager, NetApp ONTAP, and NetApp Astra Control Center.

Scenarios 1, 2 and 3 of the solution were validated using the versions as shown in the table below:

Component	Version
VMware	vSphere Client version 8.0.0.10200
	VMware ESXi, 8.0.0, 20842819
Hub Cluster	OpenShift 4.11.34

<b>Source and Destination Clusters</b>	OpenShift 4.12.9 on-premises and in AWS
<b>NetApp Astra Trident</b>	Trident Server and Client 23.04.0
<b>NetApp Astra Control Center</b>	ACC 22.11.0-82
<b>NetApp ONTAP</b>	ONTAP 9.12.1
<b>AWS FSx for NetApp ONTAP</b>	Single AZ

Scenario 4 of the solution was validated using the versions as shown in the table below:

<b>Component</b>	<b>Version</b>
<b>VMware</b>	vSphere Client version 8.0.2.00000 VMware ESXi, 8.0.2, 22380479
<b>Hub Cluster</b>	OpenShift 4.13.13
<b>Source and Destination Clusters</b>	OpenShift 4.13.12 on-premises and in Google Cloud
<b>NetApp Astra Trident</b>	Trident Server and Client 23.07.0
<b>NetApp Astra Control Center</b>	ACC 23.07.0-25
<b>NetApp ONTAP</b>	ONTAP 9.12.1
<b>Cloud Volumes ONTAP</b>	Single AZ, Single node,9.14.0

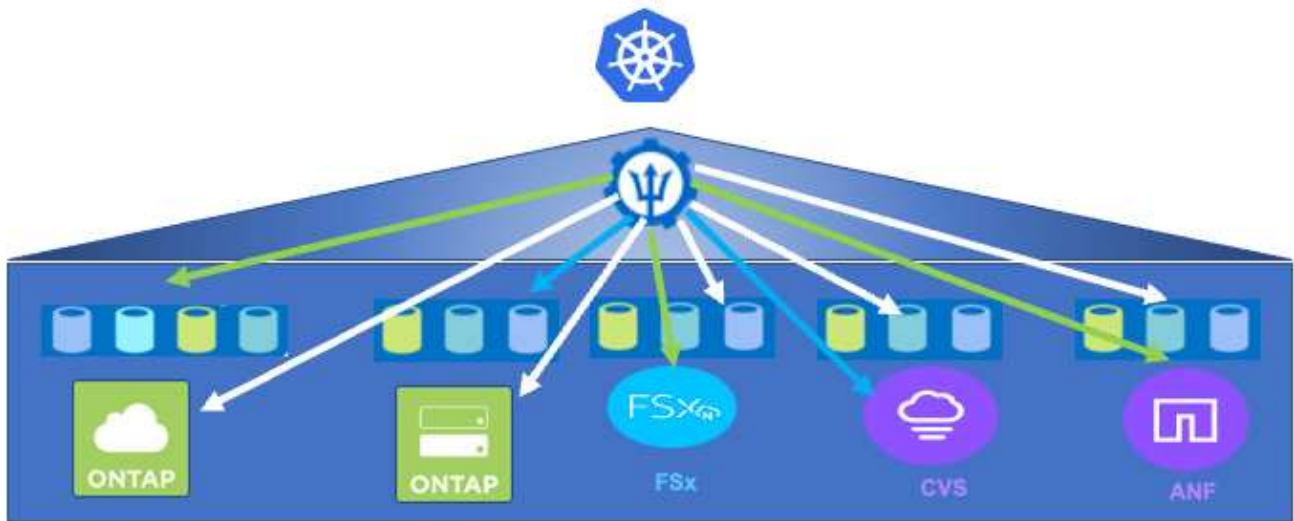
Scenario 5 of the solution was validated using the versions as shown in the table below:

<b>Component</b>	<b>Version</b>
<b>VMware</b>	vSphere Client version 8.0.2.00000 VMware ESXi, 8.0.2, 22380479
<b>Source and Destination Clusters</b>	OpenShift 4.13.25 on-premises and in Azure
<b>NetApp Astra Trident</b>	Trident Server and Client and Astra Control Provisioner 23.10.0
<b>NetApp Astra Control Center</b>	ACC 23.10
<b>NetApp ONTAP</b>	ONTAP 9.12.1
<b>Cloud Volumes ONTAP</b>	Single AZ, Single node,9.14.0

## Supported NetApp Storage integrations with Red Hat Open Shift Containers

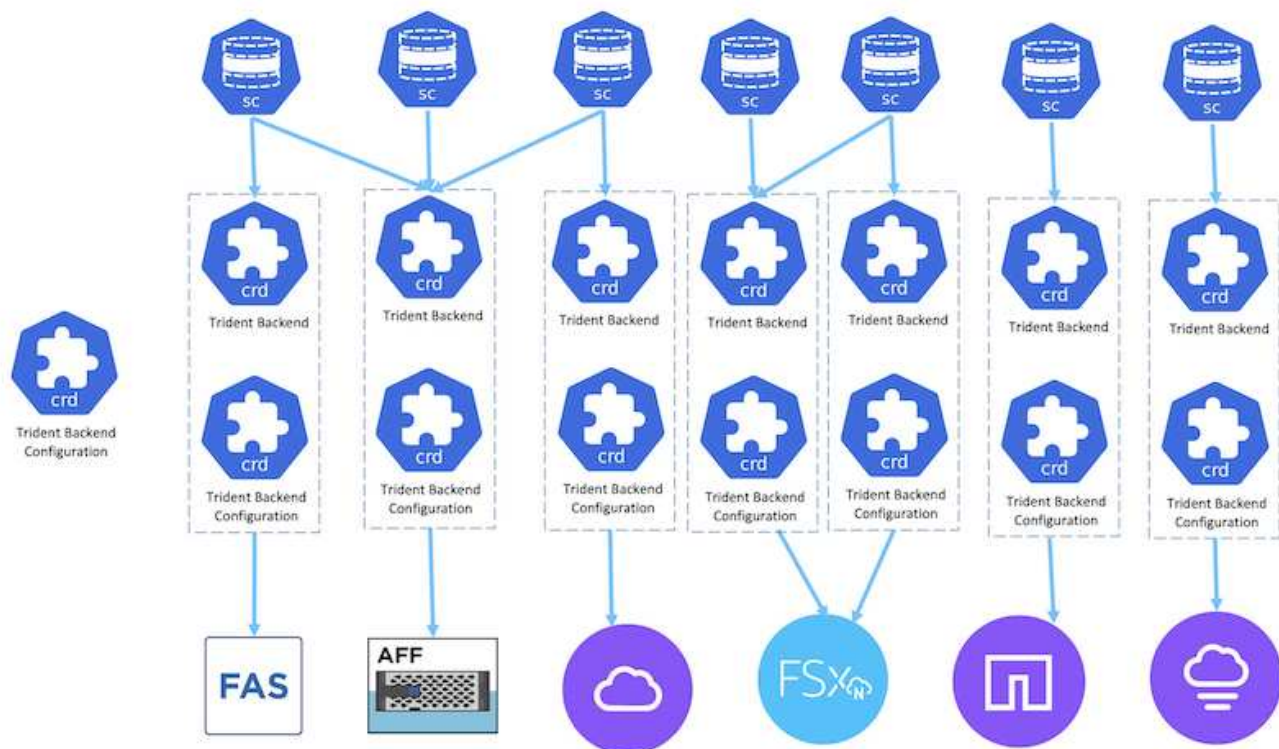
Whether the Red Hat Open Shift containers are running on VMware or in the hyperscalers, NetApp Astra Trident can be used as the CSI provisioner for the various types of backend NetApp storage that it supports.

The following diagram depicts the various backend NetApp storage that can be integrated with OpenShift clusters using NetApp Astra Trident.



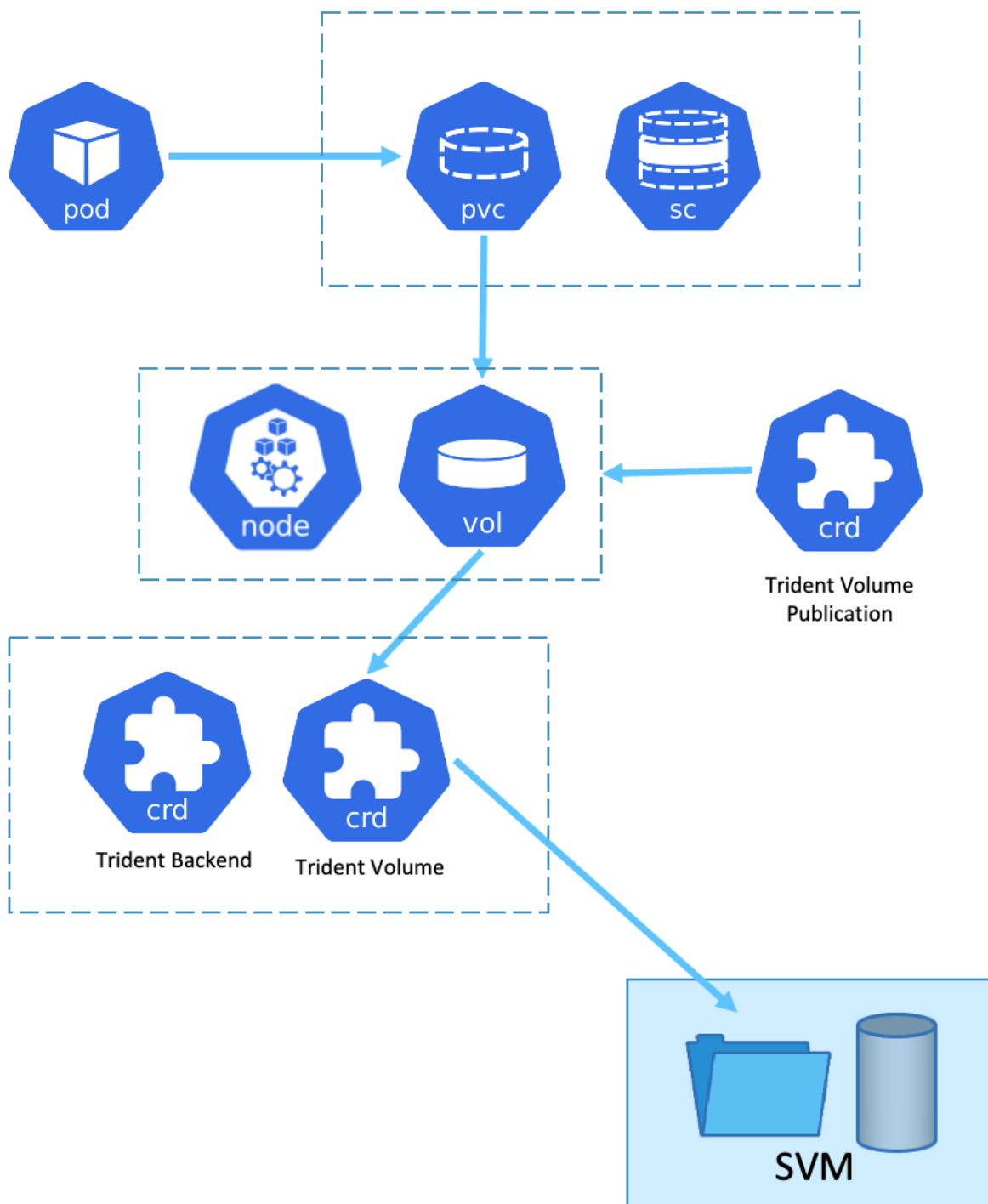
ONTAP Storage Virtual Machine (SVM) provides secure multi-tenancy. A Single OpenShift cluster can connect to single SVM or multiple SVMs or even to multiple ONTAP clusters. Storage class filters the backend storage based on parameters or by labels. Storage administrators define the parameters to connect to storage system using trident backend configuration. On successful connection establishment, it creates the trident backend and populates the information which the storage class can filter.

The relationship between the storageclass and backend is shown below.



Application owner requests persistent volume using storage class. The storage class filters the backend storage.

The relationship between the pod and backend storage is shown below.



---

## Container Storage Interface (CSI) Options

On vSphere environments, customers can pick VMware CSI driver and/or Astra Trident CSI to integrate with ONTAP. With VMware CSI, the persistent volumes are consumed as local SCSI disks, whereas with Trident, it is consumed with network.

As VMware CSI does not support RWX access modes with ONTAP, applications need to use Trident CSI if RWX mode is required. With FC based deployments, VMware CSI is preferred and SnapMirror Business Continuity (SMBC) provides zone level high availability.



## VMware CSI supports

- Core Block based datastores (FC, FCoE, iSCSI, NVMeoF)
- Core File based datastores (NFS v3, v4)
- vVol datastores (block and file)

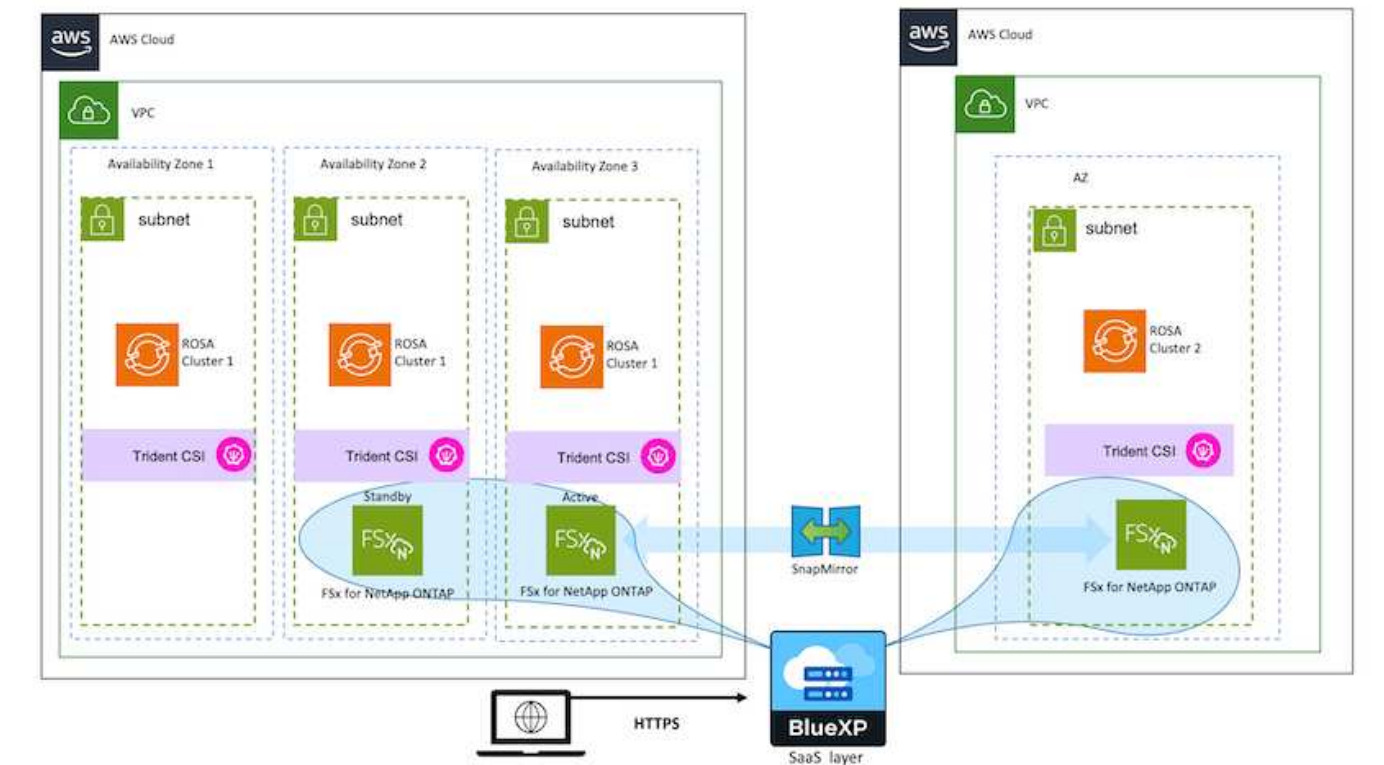
## Trident has following drivers to support ONTAP

- ontap-san (dedicated volume)
- ontap-san-economy (shared volume)
- ontap-nas (dedicated volume)
- ontap-nas-economy (shared volume)
- ontap-nas-flexgroup (dedicated large scale volume)

For both VMware CSI and Astra Trident CSI, ONTAP supports nconnect, session trunking, kerberos, etc. for NFS and multipathing, chap authentication, etc. for block protocols.

In AWS, FSx for NetApp ONTAP (FSxN) can be deployed in single Availability Zone (AZ) or in Multi AZ. For production workloads that requires high availability, multi-AZ provides zonal level fault tolerance and has better NVMe read cache compared to single AZ. For more info, check [AWS performance guidelines](#).

To save cost on disaster recovery site, single AZ FSx ONTAP can be utilized.



For number of SVMs that are supported by FSx ONTAP, refer [managing FSx ONTAP storage virtual machine](#)

## NetApp Hybrid Multicloud solutions for Red Hat OpenShift Container workloads

NetApp is seeing a significant increase in customers modernizing their legacy enterprise

applications and building new applications using containers and orchestration platforms built around Kubernetes. Red Hat OpenShift Container Platform is one example that we see adopted by many of our customers.

Overview

As more and more customers begin adopting containers within their enterprises, NetApp is perfectly positioned to help serve the persistent storage needs of their stateful applications and classic data management needs such as data protection, data security, and data migration. However, these needs are met using different strategies, tools, and methods.

**NetApp ONTAP** based storage options listed below, deliver security, data protection, reliability, and flexibility for containers and Kubernetes deployments.

- Self-managed storage in on-premises:
  - NetApp Fabric Attached Storage (FAS), NetApp All Flash FAS Arrays (AFF), NetApp All SAN Array (ASA) and ONTAP Select
- Provider-managed storage in on-premises:
  - NetApp Keystone provides Storage as a Service (STaaS)
- Self-managed storage in the cloud:
  - NetApp Cloud Volumes ONTAP(CVO) provide self managed storage in the hyperscalers
- Provider-managed storage in the cloud:
  - Cloud Volumes Service for Google Cloud (CVS), Azure NetApp Files (ANF), Amazon FSx for NetApp ONTAP offer fully managed storage in the hyperscalers

ONTAP feature highlights



<p><b>Storage Administration</b></p> <ul style="list-style-type: none"><li>• Multi-tenancy</li><li>• FlexVol &amp; FlexGroup</li><li>• LUN</li><li>• Quotas</li><li>• ONTAP CLI &amp; API</li><li>• System Manager &amp; BlueXP</li></ul>	<p><b>Performance &amp; Scalability</b></p> <ul style="list-style-type: none"><li>• FlexCache</li><li>• FlexClone</li><li>• nconnect, session trunking, multipathing</li><li>• Scale-out clusters</li></ul>
<p><b>Availability &amp; Resilience</b></p> <ul style="list-style-type: none"><li>• Multi-AZ HA deployment (MetroCluster)</li><li>• SnapShot &amp; SnapRestore</li><li>• SnapMirror</li><li>• SnapMirror Business Continuity</li><li>• SnapMirror Cloud</li></ul>	<p><b>Access Protocols</b></p> <ul style="list-style-type: none"><li>• NFS –v3, v4, v4.1, v4.2</li><li>• SMB – v2, v3</li><li>• iSCSI</li><li>• Multi-protocol access</li></ul>
<p><b>Storage Efficiency</b></p> <ul style="list-style-type: none"><li>• Deduplication &amp; Compression</li><li>• Compaction</li><li>• Thin provisioning</li><li>• Data Tiering (Fabric Pool)</li></ul>	<p><b>Security &amp; Compliance</b></p> <ul style="list-style-type: none"><li>• Fpolicy &amp; Vscan</li><li>• Active Directory integration</li><li>• LDAP &amp; Kerberos</li><li>• Certificate based authentication</li></ul>

**NetApp BlueXP** enables you to manage all of your storage and data assets from a single control plane/interface.

You can use BlueXP to create and administer cloud storage (for example, Cloud Volumes ONTAP and Azure



NetApp Files), to move, protect, and analyze data, and to control many on-prem and edge storage devices.

**NetApp Astra Trident** is a CSI Compliant Storage Orchestrator that enable quick and easy consumption of persistent storage backed by a variety of the above-mentioned NetApp storage options. It is an open-source software maintained and supported by NetApp.

## Astra Trident CSI feature highlights



<b>CSI specific</b> <ul style="list-style-type: none"><li>• CSI NetApp® Snapshot™ copies and volume creation from CSI Snapshot copies</li><li>• CSI topology</li><li>• Volume expansion</li></ul>	<b>Security</b> <ul style="list-style-type: none"><li>• Dynamic-export policy management</li><li>• iSCSI initiator-groups dynamic management</li><li>• iSCSI bidirectional CHAP</li></ul>
<b>Control</b> <ul style="list-style-type: none"><li>• Storage and performance consumption</li><li>• Monitoring</li><li>• Volume Import</li><li>• Cross Namespace Volume Access</li></ul>	<b>Installation methods</b> <ul style="list-style-type: none"><li>• Binary</li><li>• Helm chart</li><li>• Operator</li><li>• GitOps</li></ul>
<b>Choose your access mode</b> <ul style="list-style-type: none"><li>• RWO (ReadWriteOnce, i.e 1↔1)</li><li>• RWX (ReadWriteMany, i.e 1↔n)</li><li>• ROX (ReadOnlyMany)</li><li>• RWOP (ReadWriteOnce POD)</li></ul>	<b>Choose your protocol</b> <ul style="list-style-type: none"><li>• NFS</li><li>• SMB</li><li>• iSCSI</li></ul>

Business critical container workloads need more than just persistent volumes. Their data management requirements require protection and migration of the application kubernetes objects as well.



Application data includes kubernetes objects in addition to the user data: Some examples are as follows:

- kubernetes objects such as pods specs, PVCs, deployments, services
- custom config objects such as config maps and secrets
- persistent data such as Snapshot copies, backups, clones
- custom resources such as CRs and CRDs

**NetApp Astra Control**, available as both fully-managed and self-managed software, provides orchestration for robust application data management. Refer to the [Astra documentation](#) for additional details on the Astra family of products.

This reference documentation provides validation of migration and protection of container-based applications, deployed on RedHat OpenShift container platform, using NetApp Astra Control Center. In addition, the solution provides high-level details for the deployment and the use of Red Hat Advanced Cluster Management (ACM) for managing the container platforms. The document also highlights the details for the integration of NetApp storage with Red Hat OpenShift container platforms using Astra Trident CSI provisioner. Astra Control Center is deployed on the hub cluster and is used to manage the container applications and their persistent storage lifecycle. Finally, it provides a solution for replication and failover and fail-back for container workloads on managed Red Hat OpenShift clusters in AWS (ROSA) using Amazon FSx for NetApp ONTAP (FSxN) as persistent storage.

## NetApp Solution with Red Hat OpenShift Container platform workloads on VMware

If customers have a need to run their modern containerized applications on infrastructure in their private data centers, they can do so. They should plan and deploy the Red Hat OpenShift container platform (OCP) for a successful production-ready environment for deploying their container workloads. Their OCP clusters can be deployed on VMware or bare metal.

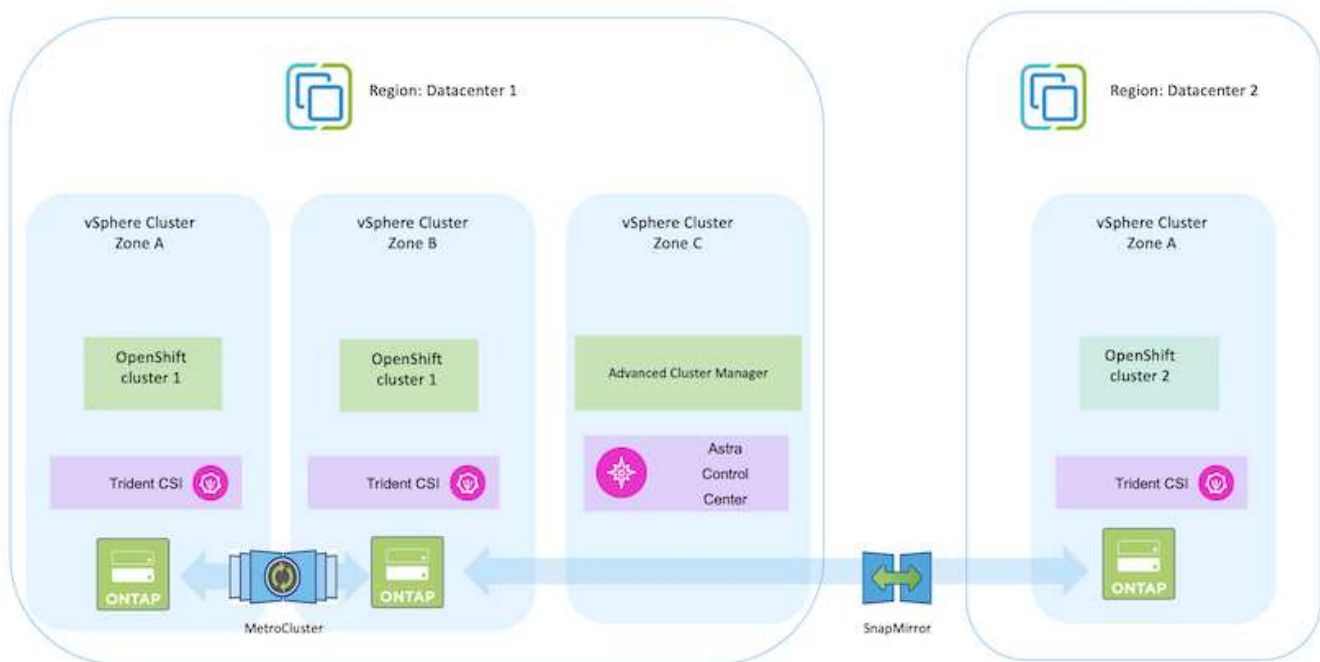
NetApp ONTAP storage delivers data protection, reliability, and flexibility for container deployments. Astra Trident serves as the dynamic storage provisioner to consume persistent ONTAP storage for customers' stateful applications. Astra Control Center can be used to orchestrate the many data management requirements of stateful applications such as data protection, migration, and business continuity.

With VMware vSphere, NetApp ONTAP tools provides a vCenter Plugin which can be utilized to provision datastores. Apply tags and use it with OpenShift for storing the node configuration and data. NVMe based storage provides lower latency and high performance.

This solution provides details for data protection and migration of container workloads using Astra Control Center. For this solution, the container workloads are deployed on Red Hat OpenShift clusters on vSphere within the on-premises environment.

NOTE: We will provide a solution for container workloads on OpenShift clusters on bare metal in the future.

### Data protection and migration solution for OpenShift Container workloads using Astra Control Center



## Deploy and configure the Red Hat OpenShift Container platform on VMware

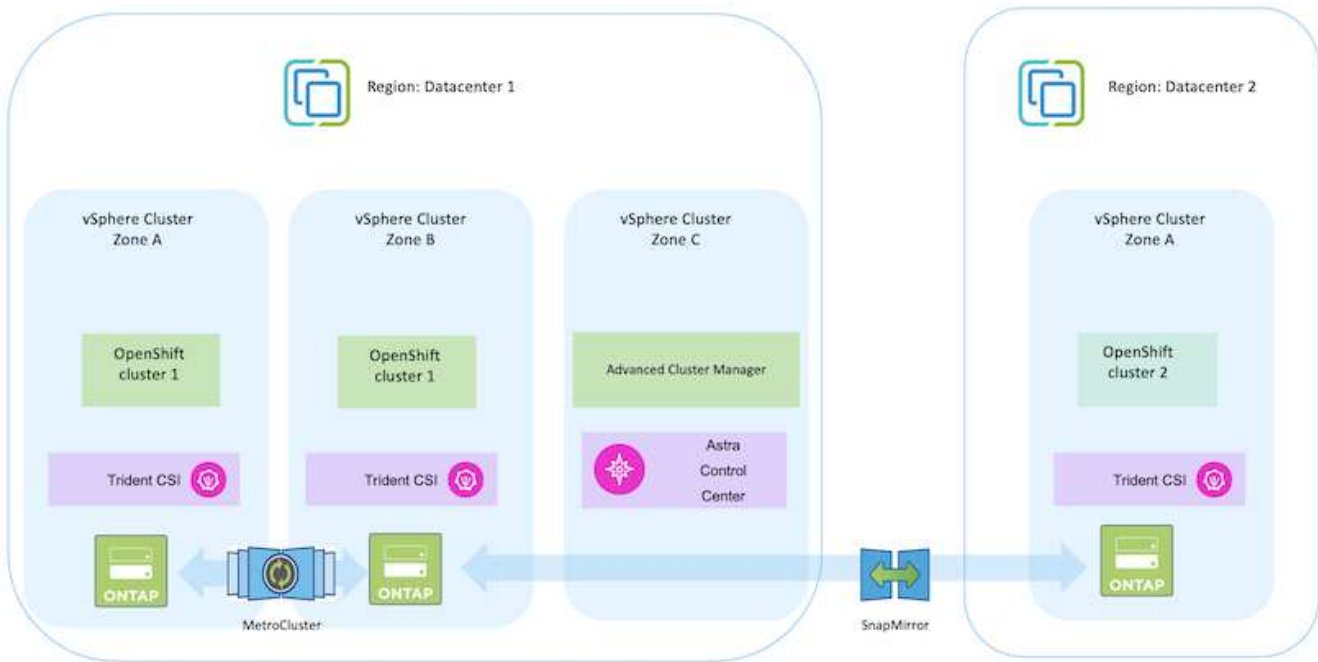
This section describes a high-level workflow of how to set up and manage OpenShift clusters and manage stateful applications on them. It shows the use of NetApp ONTAP storage arrays with the help of Astra Trident to provide persistent volumes. Details are

provided about the use of Astra Control Center to perform data protection and migration activities for the stateful applications.



There are several ways of deploying Red Hat OpenShift Container platform clusters. This high-level description of the setup provides documentation links for the specific method that was used. You can refer to the other methods in the relevant links provided in the [resources section](#).

Here is a diagram that depicts the clusters deployed on VMware in a data center.



The setup process can be broken down into the following steps:

#### Deploy and configure a CentOS VM

- It is deployed in the VMware vSphere environment.
- This VM is used for deploying some components such as NetApp Astra Trident and NetApp Astra Control Center for the solution.
- A root user is configured on this VM during installation.

## Deploy and configure an OpenShift Container Platform cluster on VMware vSphere (Hub Cluster)

Refer to the instructions for the [Assisted deployment](#) method to deploy an OCP cluster.



Remember the following:

- Create ssh public and private key to provide to the installer. These keys will be used to login to the master and worker nodes if needed.
- Download the installer program from the assisted installer. This program is used to boot the VMs that you create in the VMware vSphere environment for the master and worker nodes.
- VMs should have the minimum CPU, memory, and hard disk requirement. (Refer to the vm create commands on [this](#) page for the master and the worker nodes which provide this information)
- The diskUUID should be enabled on all VMs.
- Create a minimum of 3 nodes for master and 3 nodes for worker.
- Once they are discovered by the installer, turn on the VMware vSphere integration toggle button.

### Install Advanced Cluster Management on the Hub cluster

This is installed using the Advanced Cluster Management Operator on the Hub Cluster.  
Refer to the instructions [here](#).

### Install an internal Red Hat Quay registry on the Hub Cluster.

- An internal registry is required to push the Astra image. A Quay internal registry is installed using the Operator in the Hub cluster.
- Refer to the instructions [here](#)

### Install two additional OCP clusters (Source and Destination)

- The additional clusters can be deployed using the ACM on the Hub Cluster.
- Refer to the instructions [here](#).

### Configure NetApp ONTAP storage

- Install an ONTAP cluster with connectivity to the OCP VMs in VMWare environment.
- Create an SVM.
- Configure NAS data lif to access the storage in SVM.

## Install NetApp Trident on the OCP clusters

- Install NetApp Trident on all three clusters: Hub, source, and destination clusters
- Refer to the instructions [here](#).
- Create a storage backend for ontap-nas .
- Create a storage class for ontap-nas.
- Refer to instructions [here](#).

## Install NetApp Astra Control Center

- NetApp Astra Control Center is installed using the Astra Operator on the Hub Cluster.
- Refer to the instructions [here](#).

Points to remember:

- \* Download NetApp Astra Control Center image from the support site.
- \* Push the image to an internal registry.
- \* Refer to instructions [here](#).

## Deploy an Application on Source Cluster

Use OpenShift GitOps to deploy an application. (eg. Postgres, Ghost)

## Add the Source and Destination clusters into Astra Control Center.

After you add a cluster to Astra Control management, you can install apps on the cluster (outside of Astra Control) and then go to the Applications page in Astra Control to define the apps and their resources. Refer to [Start managing apps section of Astra Control Center](#).

The next step is to use the Astra Control Center for Data protection and Data migration from the source to the destination cluster.

## Data protection using Astra

This page shows the data protection options for Red Hat OpenShift Container based applications running on VMware vSphere using Astra Control Center (ACC).

As users take their journey of modernizing their applications with Red Hat OpenShift, a data protection strategy should be in place to protect them from accidental deletion or any other human errors. Often a protection strategy is also required for regulatory or compliance purposes to protect their data from a disaster.

The requirements of data protection varies from reverting back to a point in time copy to automatically failing over to a different fault domain without any human intervention. Many customers pick ONTAP as their preferred storage platform for their Kubernetes applications because of its rich features like multitenancy, multi-protocol, high performance and capacity offerings, replication and caching for multi-site locations, security and flexibility.

Data protection in ONTAP can be achieved using ad-hoc or policy controlled

### - Snapshot

## - backup and restore

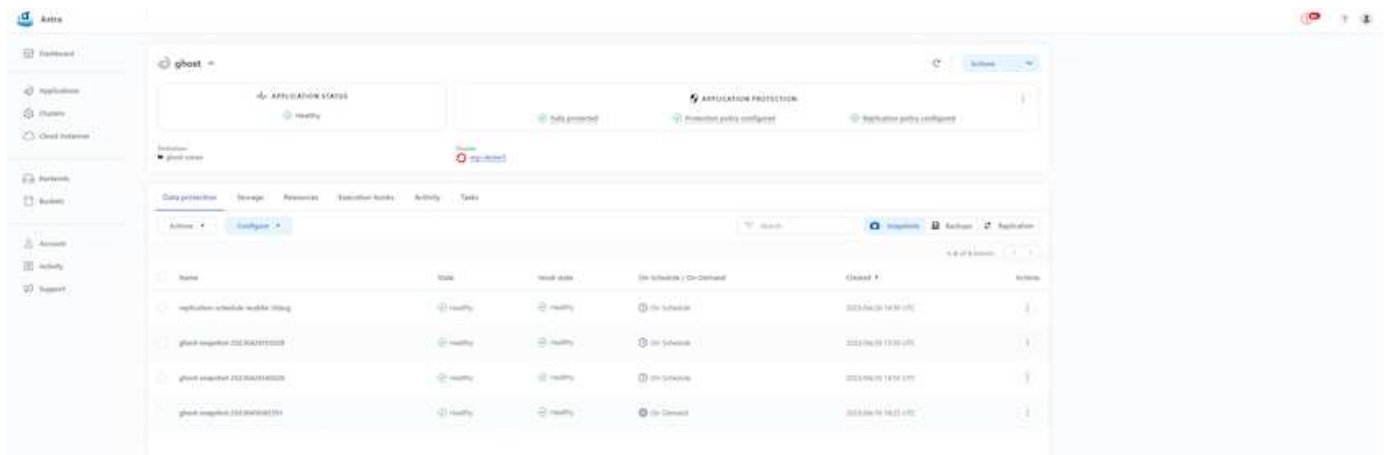
Both Snapshot copies and backups protect the following types of data:

- **The application metadata that represents the state of the application**
- **Any persistent data volumes associated with the application**
- **Any resource artifacts belonging to the application**

## Snapshot with ACC

A point in time copy of data can be captured using Snapshot with ACC. Protection policy defines the number of Snapshot copies to keep. Minimum schedule option available is hourly. Manual, on-demand Snapshot copies can be taken at any time and at shorter intervals than scheduled Snapshot copies. Snapshot copies are stored on the same provisioned volume as the app.

## Configuring Snapshot with ACC

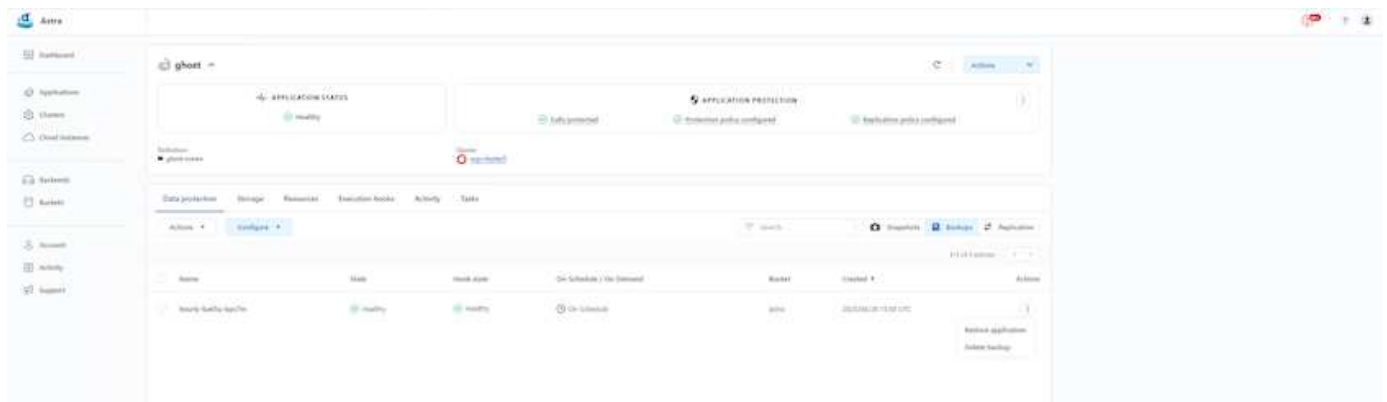


## Backup and Restore with ACC

A backup is based on a Snapshot. ACC can take Snapshot copies using CSI and perform backup using the point in time Snapshot copy. The backup is stored in an external object store (any s3 compatible including ONTAP S3 at a different location). Protection policy can be configured for scheduled backups and the number of backup versions to keep. The minimum RPO is one hour.

## Restoring an application from a backup using ACC

ACC restores application from the S3 bucket where the backups are store.



## Application specific execution hooks

In addition, execution hooks can be configured to run in conjunction with a data protection operation of a managed app. Even though storage array level data protection features are available, often additional steps are needed to make backups and restores, application consistent. The app-specific additional steps could be:

- before or after a Snapshot copy is created.
- before or after a backup is created.
- after restoring from a Snapshot copy or backup.

Astra Control can execute these app-specific steps coded as custom scripts called execution hooks.

[NetApp Verda GitHub project](#) provides execution hooks for popular cloud-native applications to make protecting applications straightforward, robust, and easy to orchestrate. Feel free to contribute to that project if you have enough information for an application that is not in the repository.

### Sample execution hook for pre-Snapshot of a redis application.

**Edit execution hook**

**HOOK DETAILS**

Operation: Pre-snapshot

Hook arguments (optional): pre

Hook name: redis-pre-snapshot

**CONTAINER IMAGES**

☐ Apply to all container images

Use a regular expression to target container images for the hook.

Container image names to match: redis

**SCRIPT**

+ Add

Search

Name
<input type="radio"/> mariadb_mysql.sh
<input type="radio"/> postgresql.sh
<input checked="" type="radio"/> redis_hook.sh

**EXECUTION HOOKS**

Execution hooks allow Astra Control to execute your own custom scripts before or after a snapshot.

Read more in [Manage application execution hooks](#)

Cancel Save

## Replication with ACC

For regional protection or for a low RPO and RTO solution, an application can be replicated to another Kubernetes instance running at a different site, preferably in another region. ACC utilizes ONTAP async SnapMirror with RPO as low as 5 minutes. Replication is done by replicating to ONTAP and then a fail over

creates the Kubernetes resources in the destination cluster.



Note that replication is different from the backup and restore where the backup goes to S3 and restore is performed from S3. Refer [xref:./rhhc/ here](#) to get additional details about the differences between the two types of data protection.

Refer [here](#) for SnapMirror setup instructions.

## SnapMirror with ACC



san-economy and nas-economy storage drivers do not support replication feature. Refer [here](#) for additional details.

## Demo video:

[Demonstration video of disaster recovery with Astra Control Center](#)

[Data protection with Astra Control Center](#)

## Business Continuity with MetroCluster

Most of our hardware platform for ONTAP has high availability features to protect from device failures avoiding the need to perform disaster recovery. But to protect from fire or any other disaster and to continue the business with zero RPO and low RTO, often a MetroCluster solution is used.

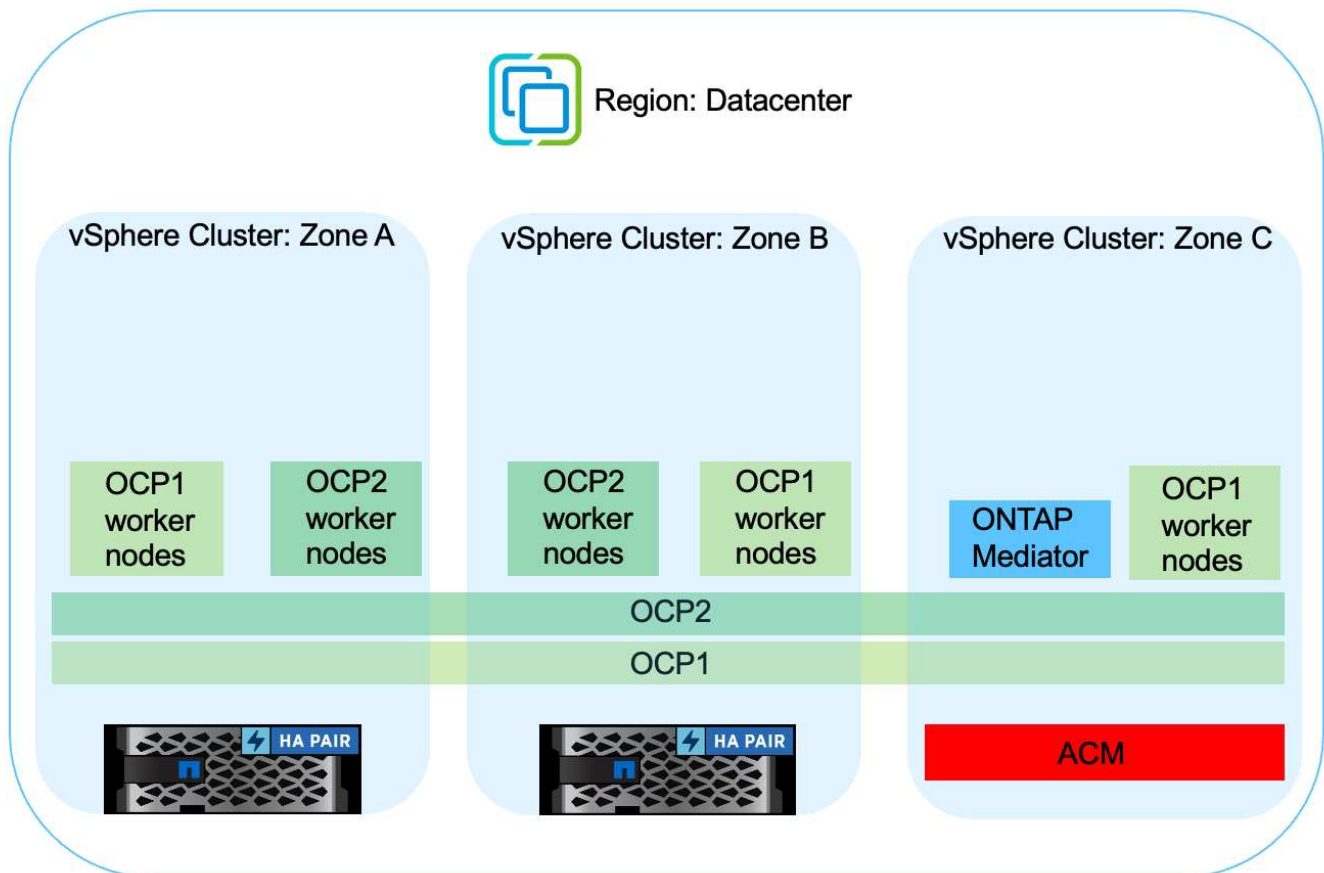
Customers who currently have an ONTAP system can extend to MetroCluster by adding supported ONTAP systems within the distance limitations for providing zone level disaster recovery.

Astra Trident, the CSI (Container Storage Interface) supports NetApp ONTAP including MetroCluster configuration as well as other options like Cloud Volumes ONTAP, Azure NetApp Files, AWS FSx for NetApp ONTAP, etc. Astra Trident provides five storage driver options for ONTAP and all are supported for MetroCluster configuration. Refer [here](#) for additional details about ONTAP storage drivers supported by Astra



Trident.

The MetroCluster solution requires layer 2 network extension or capability to access the same network address from both fault domains. Once MetroCluster configuration is in place, the solution is transparent to application owners as all the volumes in the MetroCluster svm are protected and get the benefits of SyncMirror (zero RPO).



For Trident Backend Configuration (TBC), do not specify the dataLIF and SVM when using MetroCluster configuration. Specify SVM management IP for managementLIF and use vsadmin role credentials.

Details on Astra Control Center Data Protection features are available [here](#)

## Data migration using Astra Control Center

This page shows the data migration options for container workloads on Red Hat OpenShift clusters with Astra Control Center (ACC).

Kubernetes Applications are often required to be moved from one environment to another. To migrate an application along with its persistent data, NetApp ACC can be utilized.

### Data Migration between different Kubernetes environment

ACC supports various Kubernetes flavors including Google Anthos, Red Hat OpenShift, Tanzu Kubernetes Grid, Rancher Kubernetes Engine, Upstream Kubernetes, etc. For additional details, refer [here](#).

To migrate application from one cluster to another, you can use one of the following features of ACC:

- **replication**
- **backup and restore**
- **clone**

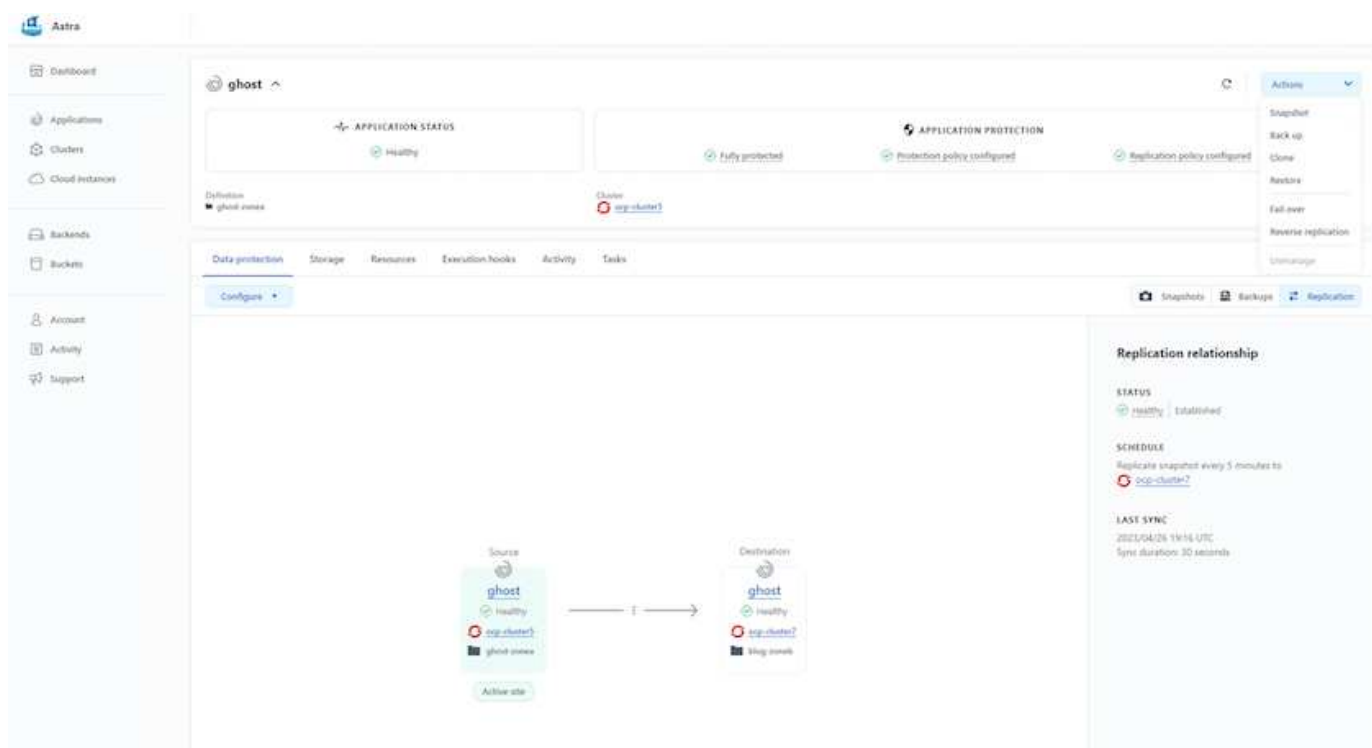
Refer to the [data protection section](#) for the **replication and backup and restore** options.

Refer [here](#) for additional details about **cloning**.



Astra Replication feature is only supported with Trident Container Storage Interface (CSI). However, replication is not supported by nas-economy & san-economy drivers.

### Performing data replication using ACC



## NetApp Hybrid Multicloud solutions for Red Hat OpenShift Container workloads

NetApp is seeing a significant increase in customers modernizing their legacy enterprise applications and building new applications using containers and orchestration platforms built around Kubernetes. Red Hat OpenShift Container Platform is one example that we see adopted by many of our customers.

### Overview

As more and more customers begin adopting containers within their enterprises, NetApp is perfectly positioned to help serve the persistent storage needs of their stateful applications and classic data management needs such as data protection, data security, and data migration. However, these needs are met using different

strategies, tools, and methods.

**NetApp ONTAP** based storage options listed below, deliver security, data protection, reliability, and flexibility for containers and Kubernetes deployments.

- Self-managed storage in on-premises:
  - NetApp Fabric Attached Storage (FAS), NetApp All Flash FAS Arrays (AFF), NetApp All SAN Array (ASA) and ONTAP Select
- Provider-managed storage in on-premises:
  - NetApp Keystone provides Storage as a Service (STaaS)
- Self-managed storage in the cloud:
  - NetApp Cloud Volumes ONTAP(CVO) provide self managed storage in the hyperscalers
- Provider-managed storage in the cloud:
  - Cloud Volumes Service for Google Cloud (CVS), Azure NetApp Files (ANF), Amazon FSx for NetApp ONTAP offer fully managed storage in the hyperscalers

## ONTAP feature highlights



<b>Storage Administration</b> <ul style="list-style-type: none"><li>• Multi-tenancy</li><li>• FlexVol &amp; FlexGroup</li><li>• LUN</li><li>• Quotas</li><li>• ONTAP CLI &amp; API</li><li>• System Manager &amp; BlueXP</li></ul>	<b>Performance &amp; Scalability</b> <ul style="list-style-type: none"><li>• FlexCache</li><li>• FlexClone</li><li>• nconnect, session trunking, multipathing</li><li>• Scale-out clusters</li></ul>
<b>Availability &amp; Resilience</b> <ul style="list-style-type: none"><li>• Multi-AZ HA deployment (MetroCluster)</li><li>• SnapShot &amp; SnapRestore</li><li>• SnapMirror</li><li>• SnapMirror Business Continuity</li><li>• SnapMirror Cloud</li></ul>	<b>Access Protocols</b> <ul style="list-style-type: none"><li>• NFS –v3, v4, v4.1, v4.2</li><li>• SMB – v2, v3</li><li>• iSCSI</li><li>• Multi-protocol access</li></ul>
<b>Storage Efficiency</b> <ul style="list-style-type: none"><li>• Deduplication &amp; Compression</li><li>• Compaction</li><li>• Thin provisioning</li><li>• Data Tiering (Fabric Pool)</li></ul>	<b>Security &amp; Compliance</b> <ul style="list-style-type: none"><li>• Fpolicy &amp; Vscan</li><li>• Active Directory integration</li><li>• LDAP &amp; Kerberos</li><li>• Certificate based authentication</li></ul>

**NetApp BlueXP** enables you to manage all of your storage and data assets from a single control plane/interface.

You can use BlueXP to create and administer cloud storage (for example, Cloud Volumes ONTAP and Azure NetApp Files), to move, protect, and analyze data, and to control many on-prem and edge storage devices.

**NetApp Astra Trident** is a CSI Compliant Storage Orchestrator that enable quick and easy consumption of persistent storage backed by a variety of the above-mentioned NetApp storage options. It is an open-source software maintained and supported by NetApp.

## Astra Trident CSI feature highlights



<b>CSI specific</b> <ul style="list-style-type: none"> <li>• CSI NetApp® Snapshot™ copies and volume creation from CSI Snapshot copies</li> <li>• CSI topology</li> <li>• Volume expansion</li> </ul>	<b>Security</b> <ul style="list-style-type: none"> <li>• Dynamic-export policy management</li> <li>• iSCSI initiator-groups dynamic management</li> <li>• iSCSI bidirectional CHAP</li> </ul>
<b>Control</b> <ul style="list-style-type: none"> <li>• Storage and performance consumption</li> <li>• Monitoring</li> <li>• Volume Import</li> <li>• Cross Namespace Volume Access</li> </ul>	<b>Installation methods</b> <ul style="list-style-type: none"> <li>• Binary</li> <li>• Helm chart</li> <li>• Operator</li> <li>• GitOps</li> </ul>
<b>Choose your access mode</b> <ul style="list-style-type: none"> <li>• RWO (ReadWriteOnce, i.e 1↔1)</li> <li>• RWX (ReadWriteMany, i.e 1↔n)</li> <li>• ROX (ReadOnlyMany)</li> <li>• RWOP (ReadWriteOnce POD)</li> </ul>	<b>Choose your protocol</b> <ul style="list-style-type: none"> <li>• NFS</li> <li>• SMB</li> <li>• iSCSI</li> </ul>

Business critical container workloads need more than just persistent volumes. Their data management requirements require protection and migration of the application kubernetes objects as well.



Application data includes kubernetes objects in addition to the user data: Some examples are as follows:

- kubernetes objects such as pods specs, PVCs, deployments, services
- custom config objects such as config maps and secrets
- persistent data such as Snapshot copies, backups, clones
- custom resources such as CRs and CRDs

**NetApp Astra Control**, available as both fully-managed and self-managed software, provides orchestration for robust application data management. Refer to the [Astra documentation](#) for additional details on the Astra family of products.

This reference documentation provides validation of migration and protection of container-based applications, deployed on RedHat OpenShift container platform, using NetApp Astra Control Center. In addition, the solution provides high-level details for the deployment and the use of Red Hat Advanced Cluster Management (ACM) for managing the container platforms. The document also highlights the details for the integration of NetApp storage with Red Hat OpenShift container platforms using Astra Trident CSI provisioner. Astra Control Center is deployed on the hub cluster and is used to manage the container applications and their persistent storage lifecycle. Finally, it provides a solution for replication and failover and fail-back for container workloads on managed Red Hat OpenShift clusters in AWS (ROSA) using Amazon FSx for NetApp ONTAP (FSxN) as persistent storage.

## NetApp Solution with Red Hat OpenShift Container platform workloads in Hybrid Cloud

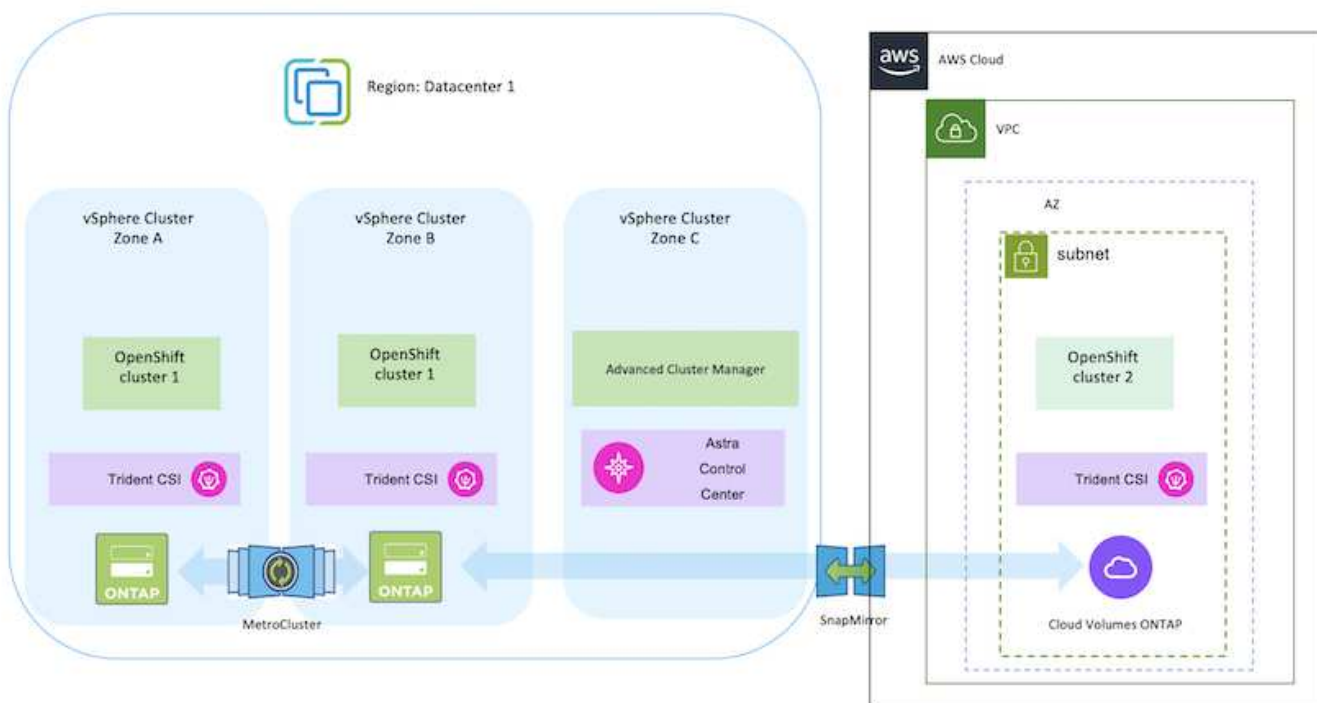
Customers may be at a point in their modernization journey when they are ready to move some select workloads or all workloads from their data centers to the cloud. They may choose to use self-managed OpenShift containers and self-managed NetApp storage in the cloud for various reasons. They should plan and deploy the Red Hat OpenShift

container platform (OCP) in the cloud for a successful production-ready environment for migrating their container workloads from their data centers. Their OCP clusters can be deployed on VMware or Bare Metal in their data centers and on AWS, Azure or Google Cloud in the cloud environment.

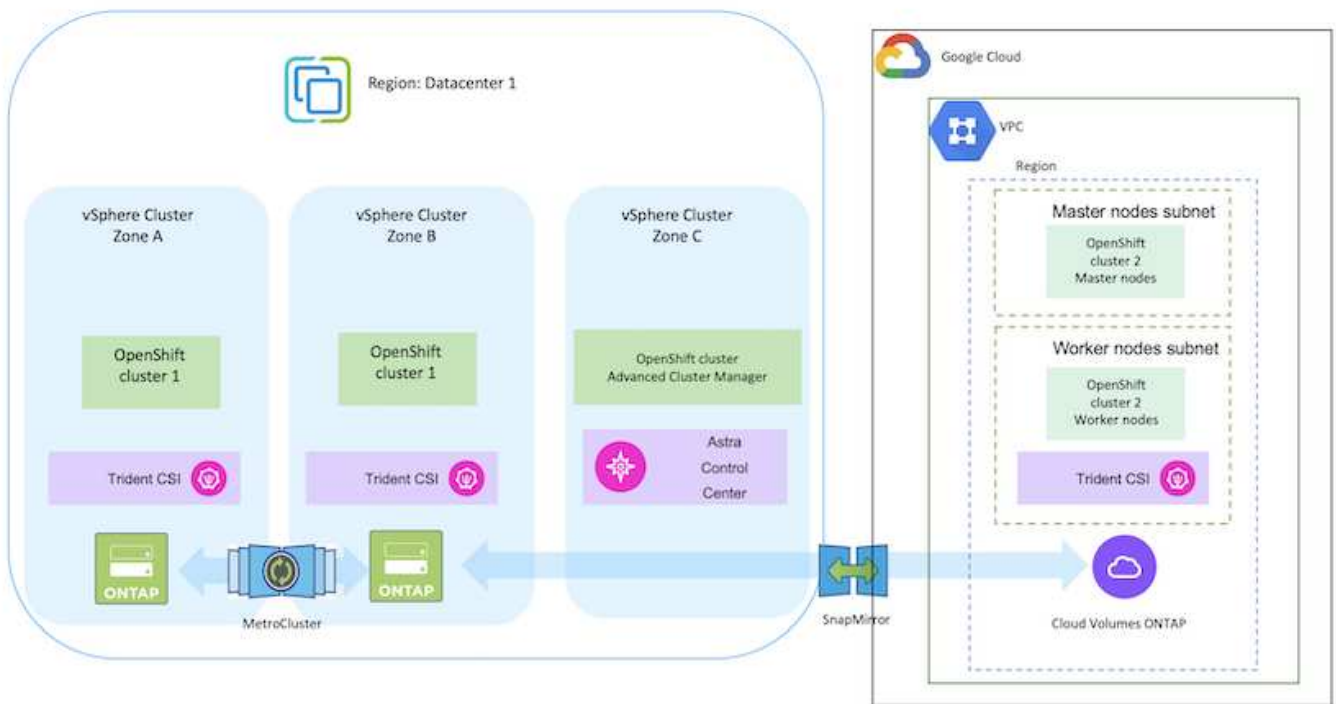
NetApp Cloud Volumes ONTAP storage delivers data protection, reliability, and flexibility for container deployments in AWS, Azure and in Google Cloud. Astra Trident serves as the dynamic storage provisioner to consume the persistent Cloud Volumes ONTAP storage for customers' stateful applications. Astra Control Center can be used to orchestrate the many data management requirements of stateful applications such as data protection, migration, and business continuity.

### Data protection and migration solution for OpenShift Container workloads in a hybrid cloud using Astra Control Center

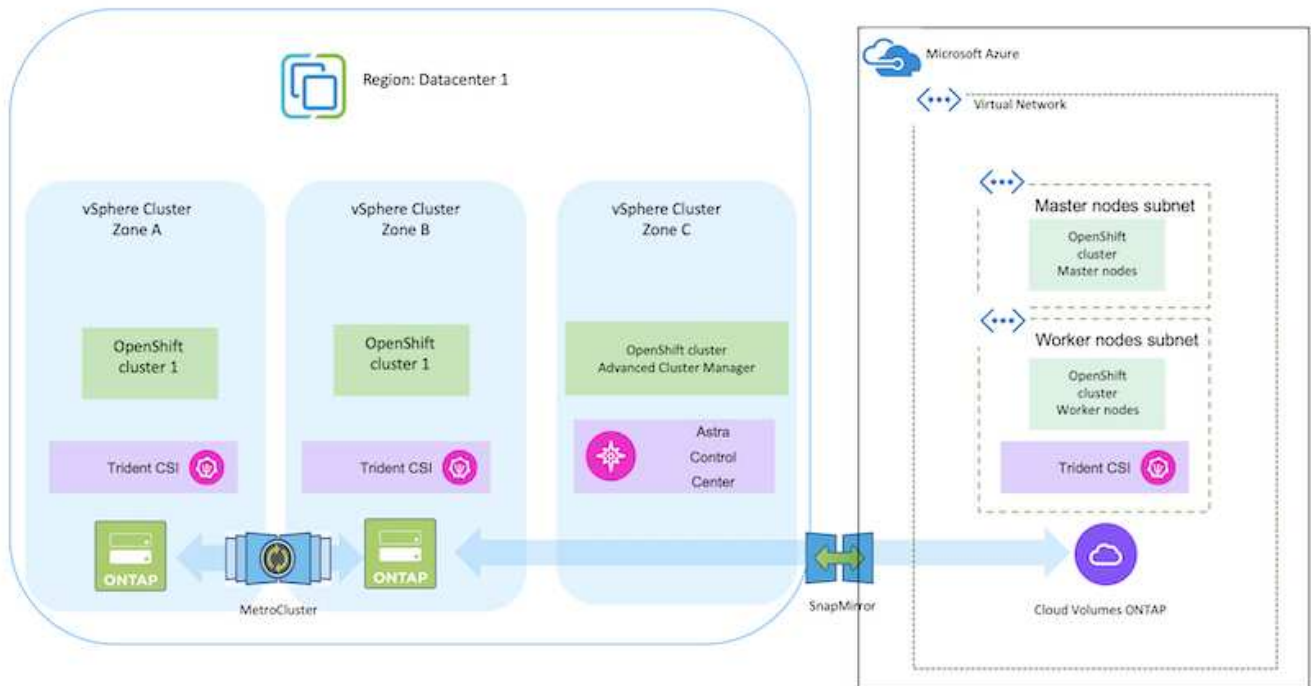
On-premises and AWS



On-premises and Google Cloud



## On-premises and Azure Cloud





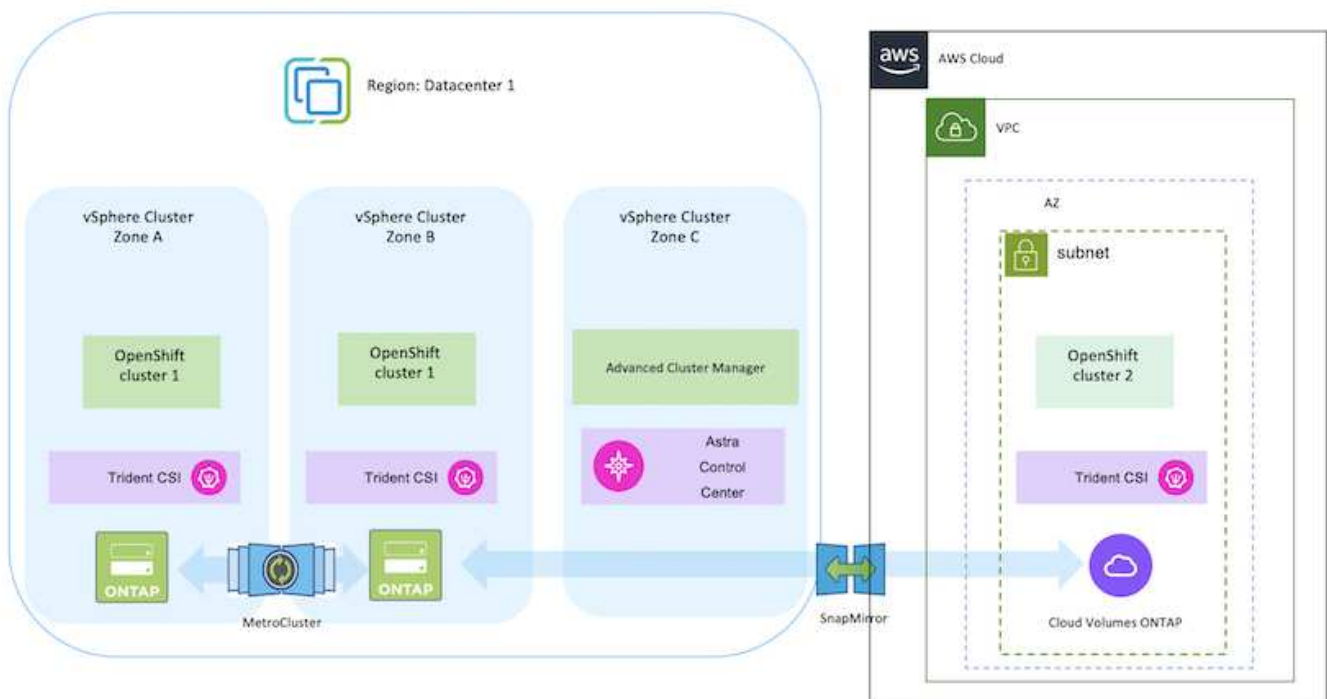
## Deploy and configure the Red Hat OpenShift Container platform on AWS

This section describes a high-level workflow of how to set up and manage OpenShift Clusters in AWS and deploy stateful applications on them. It shows the use of NetApp Cloud Volumes ONTAP storage with the help of Astra Trident to provide persistent volumes. Details are provided about the use of Astra Control Center to perform data protection and migration activities for the stateful applications.



There are several ways of deploying Red Hat OpenShift Container platform clusters on AWS. This high-level description of the setup provides documentation links for the specific method that was used. You can refer to the other methods in the relevant links provided in the [resources section](#).

Here is a diagram that depicts the clusters deployed on AWS and connected to the data center using a VPN.



The setup process can be broken down into the following steps:

## Install an OCP cluster on AWS from the Advanced Cluster Management.

- Create a VPC with a site-to-site VPN connection (using pfsense) to connect to the on-premises network.
- On-premises network has internet connectivity.
- Create 3 private subnets in 3 different AZs.
- Create a Route 53 private hosted zone and a DNS resolver for the VPC.

Create OpenShift Cluster on AWS from the Advanced Cluster Management (ACM) Wizard. Refer to instructions [here](#).



You can also create the cluster in AWS from the OpenShift Hybrid Cloud console. Refer [here](#) for instructions.



When creating the cluster using the ACM, you have the ability to customize the installation by editing the yaml file after filling in the details in the form view. After the cluster is created, you can ssh login to the nodes of the cluster for troubleshooting or additional manual configuration. Use the ssh key you provided during installation and the username core to login.

## Deploy Cloud Volumes ONTAP in AWS using BlueXP.

- Install the connector in on-premises VMware environment. Refer to instructions [here](#).
- Deploy a CVO instance in AWS using the connector. Refer to instructions [here](#).



The connector can also be installed in the cloud environment. Refer [here](#) for additional information.

## Install Astra Trident in the OCP Cluster

- Deploy Trident Operator using Helm.  
Refer to instructions [here](#)
- Create a backend and a storage class. Refer to instructions [here](#).

## Add the OCP cluster on AWS to the Astra Control Center.

Add the OCP cluster in AWS to Astra Control Center.

## Using CSI Topology feature of Trident for multi-zone architectures

Cloud providers, today, enable Kubernetes/OpenShift cluster administrators to spawn nodes of the clusters that are zone based. Nodes can be located in different availability zones within a region, or across various regions. To facilitate the provisioning of volumes for workloads in a multi-zone architecture, Astra Trident uses CSI Topology. Using the CSI Topology feature, access to volumes can be limited to a subset of nodes, based on regions and availability zones. Refer [here](#) for additional details.



Kubernetes supports two volume binding modes:

- When **VolumeBindingMode is set to Immediate** (default), Astra Trident creates the volume without any topology awareness. Persistent Volumes are created without having any dependency on the requesting pod's scheduling requirements.
- When **VolumeBindingMode set to WaitForFirstConsumer**, the creation and binding of a Persistent Volume for a PVC is delayed until a pod that uses the PVC is scheduled and created. This way, volumes are created to meet the scheduling constraints that are enforced by topology requirements.

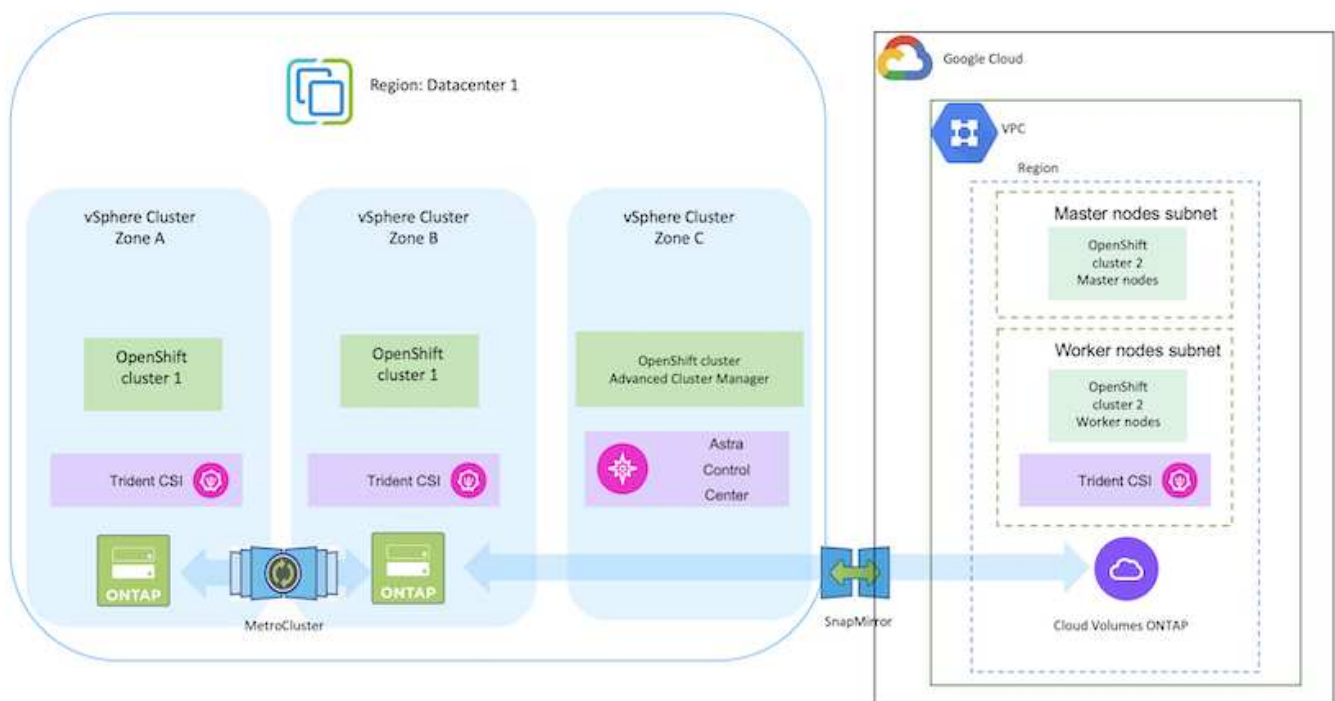
Astra Trident storage backends can be designed to selectively provision volumes based on availability zones (Topology-aware backend). For StorageClasses that make use of such a backend, a volume would only be created if requested by an application that is scheduled in a supported region/zone. (Topology-aware StorageClass)

Refer [here](#) for additional details.

## Deploy and configure the Red Hat OpenShift Container platform on GCP

This section describes a high-level workflow of how to set up and manage OpenShift Clusters in GCP and deploy stateful applications on them. It shows the use of NetApp Cloud Volumes ONTAP storage with the help of Astra Trident to provide persistent volumes. Details are provided about the use of Astra Control Center to perform data protection and migration activities for the stateful applications.

Here is a diagram that shows the clusters deployed on GCP and connected to the data center using a VPN.



There are several ways of deploying Red Hat OpenShift Container platform clusters in GCP. This high-level description of the setup provides documentation links for the specific method that was used. You can refer to the other methods in the relevant links provided in the [resources section](#).

The setup process can be broken down into the following steps:

#### **Install an OCP cluster on GCP from the CLI.**

- Ensure that you have met all the prerequisites stated [here](#).
- For the VPN connectivity between on-premises and GCP, a pfsense VM was created and configured. For instructions, see [here](#).
  - The remote gateway address in pfsense can be configured only after you have created a VPN gateway in Google Cloud Platform.
  - The remote network IP addresses for the Phase 2 can be configured only after the OpenShift cluster installation program runs and creates the infrastructure components for the cluster.
  - The VPN in Google Cloud can only be configured after the infrastructure components for the cluster are created by the installation program.
- Now install the OpenShift cluster on GCP.
  - Obtain the installation program and the pull secret and deploy the cluster following the steps provided in the documentation [here](#).
  - The installation creates a VPC network in Google Cloud Platform. It also creates a private zone in Cloud DNS and adds A records.
    - Use the CIDR block address of the VPC network to configure the pfsense and establish the VPN connection. Ensure firewalls are setup correctly.
    - Add A records in the DNS of the on-premises environment using the IP address in the A records of the Google Cloud DNS.
  - The installation of the cluster completes and will provide a kubeconfig file and username and password to login to the console of the cluster.

#### **Deploy Cloud Volumes ONTAP in GCP using BlueXP.**

- Install a connector in Google Cloud. Refer to instructions [here](#).
- Deploy a CVO instance in Google Cloud using the connector. Refer to instructions [here](https://docs.netapp.com/us-en/bluexp-cloud-volumes-ontap/task-getting-started-gcp.html).

#### **Install Astra Trident in the OCP Cluster in GCP**

- There are many methods to deploy Astra Trident as shown [here](#).
- For this project, Astra Trident was installed by deploying Astra Trident Operator manually using the instructions [here](#).
- Create backend and a storage classes. Refer to instructions [here](#).

## Add the OCP cluster on GCP to the Astra Control Center.

- Create a separate KubeConfig file with a cluster role that contains the minimum permissions necessary for a cluster to be managed by Astra Control. The instructions can be found [here](#).
- Add the cluster to Astra Control Center following the instructions [here](#)

## Using CSI Topology feature of Trident for multi-zone architectures

Cloud providers, today, enable Kubernetes/OpenShift cluster administrators to spawn nodes of the clusters that are zone based. Nodes can be located in different availability zones within a region, or across various regions. To facilitate the provisioning of volumes for workloads in a multi-zone architecture, Astra Trident uses CSI Topology. Using the CSI Topology feature, access to volumes can be limited to a subset of nodes, based on regions and availability zones. Refer [here](#) for additional details.



Kubernetes supports two volume binding modes:

- When **VolumeBindingMode is set to Immediate** (default), Astra Trident creates the volume without any topology awareness. Persistent Volumes are created without having any dependency on the requesting pod's scheduling requirements.
- When **VolumeBindingMode set to WaitForFirstConsumer**, the creation and binding of a Persistent Volume for a PVC is delayed until a pod that uses the PVC is scheduled and created. This way, volumes are created to meet the scheduling constraints that are enforced by topology requirements.

Astra Trident storage backends can be designed to selectively provision volumes based on availability zones (Topology-aware backend). For StorageClasses that make use of such a backend, a volume would only be created if requested by an application that is scheduled in a supported region/zone. (Topology-aware StorageClass)  
Refer [here](#) for additional details.

## Demonstration Video

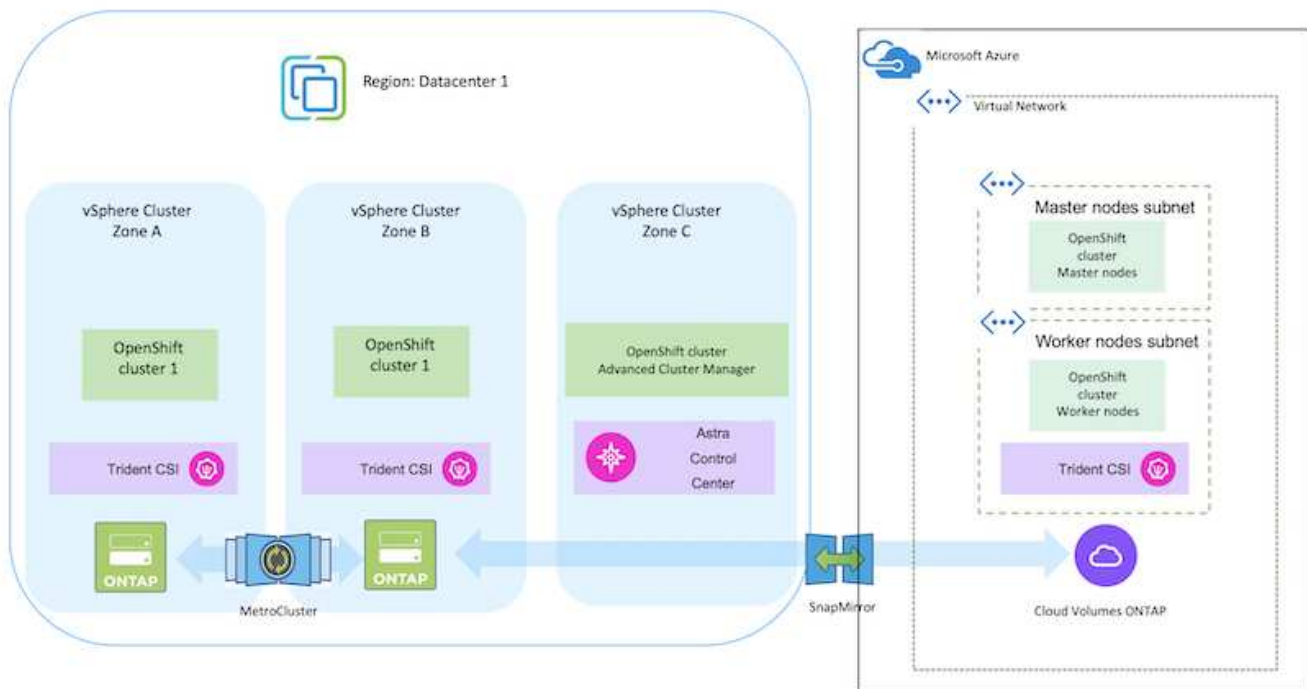
[OpenShift Cluster installation on Google Cloud Platform](#)

[Importing OpenShift clusters into Astra Control Center](#)

## Deploy and configure the Red Hat OpenShift Container platform on Azure

This section describes a high-level workflow of how to set up and manage OpenShift Clusters in Azure and deploy stateful applications on them. It shows the use of NetApp Cloud Volumes ONTAP storage with the help of Astra Trident/Astra Control Provisioner to provide persistent volumes. Details are provided about the use of Astra Control Center to perform data protection and migration activities for the stateful applications.

Here is a diagram that shows the clusters deployed on Azure and connected to the data center using a VPN.



There are several ways of deploying Red Hat OpenShift Container platform clusters in Azure. This high-level description of the setup provides documentation links for the specific method that was used. You can refer to the other methods in the relevant links provided in the [resources section](#).

The setup process can be broken down into the following steps:

## Install an OCP cluster on Azure from the CLI.

- Ensure that you have met all the prerequisites stated [here](#).
- Create a VPN, subnets and network security groups and a private DNS zone. Create VPN gateway and site-to-site VPN Connection.
- For the VPN connectivity between on-premises and Azure, a pfSense VM was created and configured. For instructions, see [here](#).
- Obtain the installation program and the pull secret and deploy the cluster following the steps provided in the documentation [here](#).
- The installation of the cluster completes and will provide a kubeconfig file and username and password to login to the console of the cluster.

A sample install-config.yaml file is given below.

```
apiVersion: v1
baseDomain: sddc.netapp.com
compute:
- architecture: amd64
  hyperthreading: Enabled
  name: worker
  platform:
    azure:
      encryptionAtHost: false
      osDisk:
        diskSizeGB: 512
        diskType: "StandardSSD_LRS"
      type: Standard_D2s_v3
      ultraSSDCapability: Disabled
      #zones:
      #- "1"
      #- "2"
      #- "3"
  replicas: 3
controlPlane:
  architecture: amd64
  hyperthreading: Enabled
  name: master
  platform:
    azure:
      encryptionAtHost: false
      osDisk:
        diskSizeGB: 1024
        diskType: Premium_LRS
      type: Standard_D8s_v3
      ultraSSDCapability: Disabled
  replicas: 3
```

```

metadata:
  creationTimestamp: null
  name: azure-cluster
networking:
  clusterNetwork:
    - cidr: 10.128.0.0/14
      hostPrefix: 23
  machineNetwork:
    - cidr: 10.0.0.0/16
  networkType: OVNKubernetes
  serviceNetwork:
    - 172.30.0.0/16
platform:
  azure:
    baseDomainResourceGroupName: ocp-base-domain-rg
    cloudName: AzurePublicCloud
    computeSubnet: ocp-subnet2
    controlPlaneSubnet: ocp-subnet1
    defaultMachinePlatform:
      osDisk:
        diskSizeGB: 1024
        diskType: "StandardSSD_LRS"
        ultraSSDCapability: Disabled
    networkResourceGroupName: ocp-nc-us-rg
    #outboundType: UserDefinedRouting
    region: northcentralus
    resourceGroupName: ocp-cluster-ncusrg
    virtualNetwork: ocp_vnet_ncus
publish: Internal
pullSecret:

```

### Deploy Cloud Volumes ONTAP in Azure using BlueXP.

- Install a connector in Azure. Refer to instructions [here](#).
- Deploy a CVO instance in Azure using the connector. Refer to instructions [link:https://docs.netapp.com/us-en/bluexp-cloud-volumes-ontap/task-getting-started-azure.html](https://docs.netapp.com/us-en/bluexp-cloud-volumes-ontap/task-getting-started-azure.html) [here.]

### Install Astra Control Provisioner in the OCP Cluster in Azure

- For this project, Astra Control Provisioner (ACP) was installed on all the clusters (on-prem cluster, on-prem cluster where Astra Control Center is deployed and the cluster in Azure). Learn more about the Astra Control Provisioner [here](#).
- Create backend and a storage classes. Refer to instructions [here](#).

## Add the OCP cluster on Azure to the Astra Control Center.

- Create a separate KubeConfig file with a cluster role that contains the minimum permissions necessary for a cluster to be managed by Astra Control. The instructions can be found [here](#).
- Add the cluster to Astra Control Center following the instructions [here](#)

## Using CSI Topology feature of Trident for multi-zone architectures

Cloud providers, today, enable Kubernetes/OpenShift cluster administrators to spawn nodes of the clusters that are zone based. Nodes can be located in different availability zones within a region, or across various regions. To facilitate the provisioning of volumes for workloads in a multi-zone architecture, Astra Trident uses CSI Topology. Using the CSI Topology feature, access to volumes can be limited to a subset of nodes, based on regions and availability zones. Refer [here](#) for additional details.



Kubernetes supports two volume binding modes:

- When **VolumeBindingMode is set to Immediate** (default), Astra Trident creates the volume without any topology awareness. Persistent Volumes are created without having any dependency on the requesting pod's scheduling requirements.
- When **VolumeBindingMode set to WaitForFirstConsumer**, the creation and binding of a Persistent Volume for a PVC is delayed until a pod that uses the PVC is scheduled and created. This way, volumes are created to meet the scheduling constraints that are enforced by topology requirements.

Astra Trident storage backends can be designed to selectively provision volumes based on availability zones (Topology-aware backend). For StorageClasses that make use of such a backend, a volume would only be created if requested by an application that is scheduled in a supported region/zone. (Topology-aware StorageClass)  
Refer [here](#) for additional details.

## Demonstration Video

[Using Astra Control for Failover and Failback of applications](#)

## Data protection using Astra Control Center

This page shows the data protection options for Red Hat OpenShift Container based applications running on VMware vSphere or in the cloud using Astra Control Center (ACC).

As users take their journey of modernizing their applications with Red Hat OpenShift, a data protection strategy should be in place to protect them from accidental deletion or any other human errors. Often a protection strategy is also required for regulatory or compliance purposes to protect their data from a disaster.

The requirements of data protection varies from reverting back to a point in time copy to automatically failing over to a different fault domain without any human intervention. Many customers pick ONTAP as their preferred storage platform for their Kubernetes applications because of its rich features like multitenancy, multi-protocol, high performance and capacity offerings, replication and caching for multi-site locations, security and flexibility.

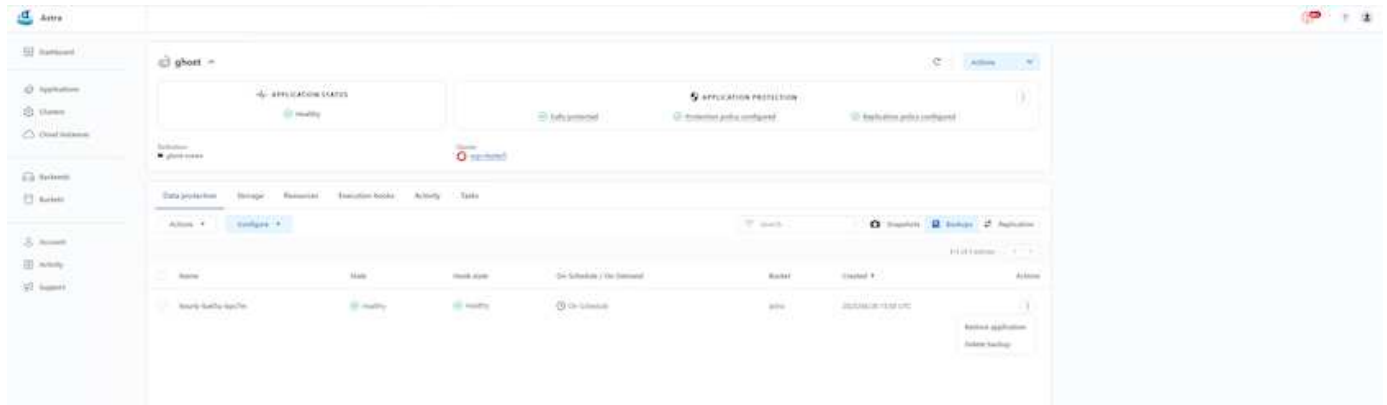
Customers may have a cloud environment setup as their data center extension, so that they can leverage the benefits of the cloud as well as be well positioned to move their workloads at a future time. For such customers, backing up of their OpenShift applications and their data to the cloud environment becomes an

inevitable choice. They can then restore the applications and the associated data either to an OpenShift cluster in the cloud or in their data center.

## Backup and Restore with ACC

Application owners can review and update the applications discovered by ACC. ACC can take Snapshot copies using CSI and perform backup using the point in time Snapshot copy. Backup destination can be an object store in the cloud environment. Protection policy can be configured for scheduled backups and the number of backup versions to keep. The minimum RPO is one hour.

### Restoring an application from a backup using ACC



### Application specific execution hooks

Even though storage array level data protection features are available, often additional steps are needed to make backups and restores application consistent. The app-specific additional steps could be:

- before or after a Snapshot copy is created.
- before or after a backup is created.
- after restoring from a Snapshot copy or backup.

Astra Control can execute these app-specific steps coded as custom scripts called execution hooks.

NetApp's [open source project Verda](#) provides execution hooks for popular cloud-native applications to make protecting applications straightforward, robust, and easy to orchestrate. Feel free to contribute to that project if you have enough information for an application that is not in the repository.

### Sample execution hook for pre-Snapshot of a redis application.



Edit execution hook

HOOK DETAILS ?

Operation

Pre-snapshot

Hook arguments (optional)

1 pre X

Enter hook arguments

Hook name

redis-pre-snapshot

CONTAINER IMAGES ?

☐ Apply to all container images

Use a regular expression to target container images for the hook.

Container image names to match

redis

SCRIPT ?

+ Add

Search

Name ↓

☐ mariadb\_mysql.sh

☐ postgresql.sh

☒ redis\_hook.sh

EXECUTION HOOKS

Execution hooks allow Astra Control to execute your own custom scripts before or after a snapshot.

Read more in [Manage application execution hooks](#)

Cancel

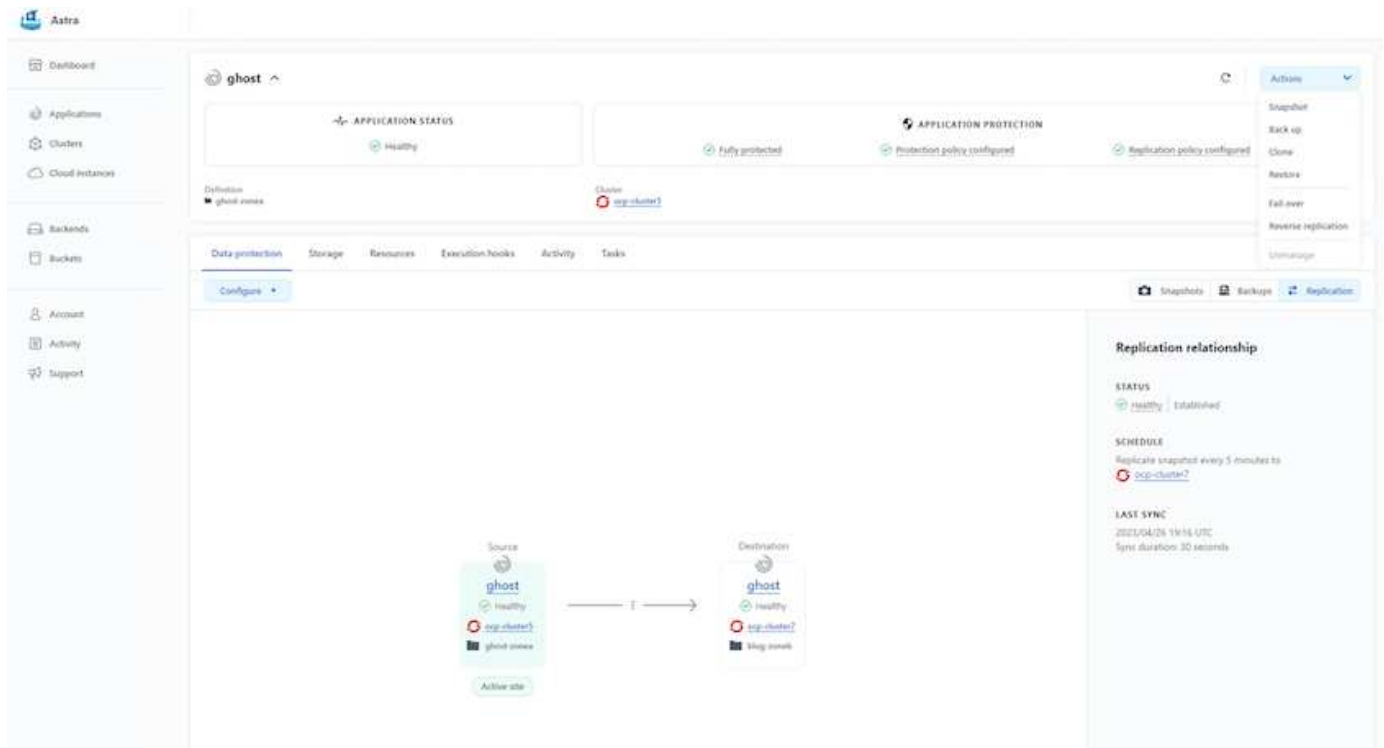
Save ✓

## Replication with ACC

For regional protection or for a low RPO and RTO solution, an application can be replicated to another Kubernetes instance running at a different site, preferably in another region. ACC utilizes ONTAP async SnapMirror with RPO as low as 5 minutes. Refer [here](#) for SnapMirror setup instructions.

## SnapMirror with ACC

39



san-economy and nas-economy storage drivers do not support replication feature. Refer [here](#) for additional details.

#### Demo video:

[Demonstration video of disaster recovery with Astra Control Center](#)

[Data protection with Astra Control Center](#)

Details on Astra Control Center Data Protection features are available [here](#)

#### Disaster recovery (Failover and Failback using replication) with ACC

[Using Astra Control for Failover and Failback of applications](#)

### Data migration using Astra Control Center

This page shows the data migration options for container workloads on Red Hat OpenShift clusters with Astra Control Center (ACC). Specifically, customers can use ACC to

- move some selected workloads or all workloads from their on-premises data centers to the cloud
- clone their apps to the cloud either for testing purposes or move from the data center to the cloud

#### Data Migration

To migrate application from one environment to another, you can use one of the following features of ACC:

- replication
- backup and restore
- clone

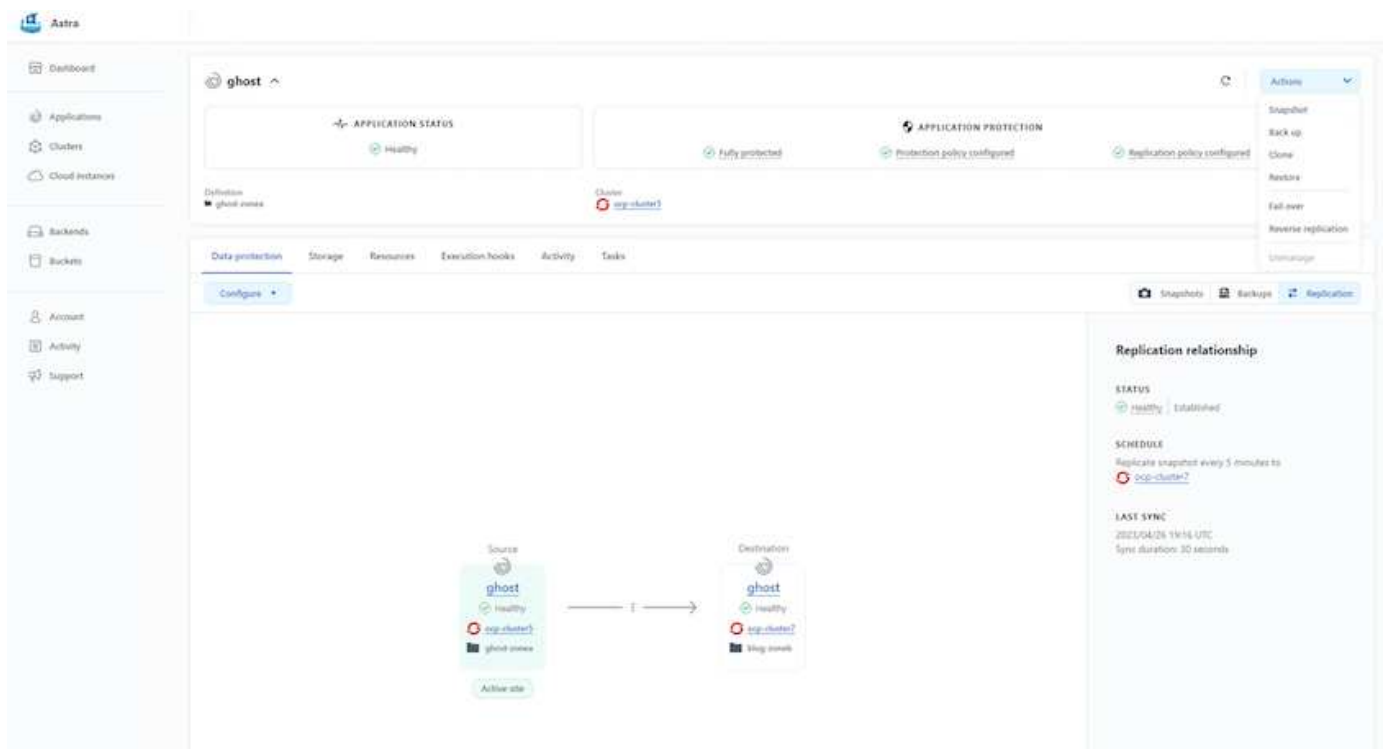
Refer to the [data protection section](#) for the **replication and backup and restore** options.

Refer [here](#) for additional details about **cloning**.



Astra Replication feature is only supported with Trident Container Storage Interface (CSI). However, replication is not supported by nas-economy & san-economy drivers.

### Performing data replication using ACC



## NetApp Hybrid Multicloud solutions for Red Hat OpenShift Container workloads

NetApp is seeing a significant increase in customers modernizing their legacy enterprise applications and building new applications using containers and orchestration platforms built around Kubernetes. Red Hat OpenShift Container Platform is one example that we see adopted by many of our customers.

### Overview

As more and more customers begin adopting containers within their enterprises, NetApp is perfectly positioned to help serve the persistent storage needs of their stateful applications and classic data management needs such as data protection, data security, and data migration. However, these needs are met using different strategies, tools, and methods.

**NetApp ONTAP** based storage options listed below, deliver security, data protection, reliability, and flexibility for containers and Kubernetes deployments.

- Self-managed storage in on-premises:
  - NetApp Fabric Attached Storage (FAS), NetApp All Flash FAS Arrays (AFF), NetApp All SAN Array (ASA) and ONTAP Select
- Provider-managed storage in on-premises:
  - NetApp Keystone provides Storage as a Service (STaaS)
- Self-managed storage in the cloud:
  - NetApp Cloud Volumes ONTAP(CVO) provide self managed storage in the hyperscalers
- Provider-managed storage in the cloud:
  - Cloud Volumes Service for Google Cloud (CVS), Azure NetApp Files (ANF), Amazon FSx for NetApp ONTAP offer fully managed storage in the hyperscalers

## ONTAP feature highlights



<b>Storage Administration</b> <ul style="list-style-type: none"> <li>Multi-tenancy</li> <li>FlexVol &amp; FlexGroup</li> <li>LUN</li> <li>Quotas</li> <li>ONTAP CLI &amp; API</li> <li>System Manager &amp; BlueXP</li> </ul>	<b>Performance &amp; Scalability</b> <ul style="list-style-type: none"> <li>FlexCache</li> <li>FlexClone</li> <li>nconnect, session trunking, multipathing</li> <li>Scale-out clusters</li> </ul>
<b>Availability &amp; Resilience</b> <ul style="list-style-type: none"> <li>Multi-AZ HA deployment (MetroCluster)</li> <li>SnapShot &amp; SnapRestore</li> <li>SnapMirror</li> <li>SnapMirror Business Continuity</li> <li>SnapMirror Cloud</li> </ul>	<b>Access Protocols</b> <ul style="list-style-type: none"> <li>NFS –v3, v4, v4.1, v4.2</li> <li>SMB – v2, v3</li> <li>iSCSI</li> <li>Multi-protocol access</li> </ul>
<b>Storage Efficiency</b> <ul style="list-style-type: none"> <li>Deduplication &amp; Compression</li> <li>Compaction</li> <li>Thin provisioning</li> <li>Data Tiering (Fabric Pool)</li> </ul>	<b>Security &amp; Compliance</b> <ul style="list-style-type: none"> <li>Fpolicy &amp; Vscan</li> <li>Active Directory integration</li> <li>LDAP &amp; Kerberos</li> <li>Certificate based authentication</li> </ul>

**NetApp BlueXP** enables you to manage all of your storage and data assets from a single control plane/interface.

You can use BlueXP to create and administer cloud storage (for example, Cloud Volumes ONTAP and Azure NetApp Files), to move, protect, and analyze data, and to control many on-prem and edge storage devices.

**NetApp Astra Trident** is a CSI Compliant Storage Orchestrator that enable quick and easy consumption of persistent storage backed by a variety of the above-mentioned NetApp storage options. It is an open-source software maintained and supported by NetApp.



## Astra Trident CSI feature highlights

<b>CSI specific</b> <ul style="list-style-type: none"> <li>• CSI NetApp® Snapshot™ copies and volume creation from CSI Snapshot copies</li> <li>• CSI topology</li> <li>• Volume expansion</li> </ul>	<b>Security</b> <ul style="list-style-type: none"> <li>• Dynamic-export policy management</li> <li>• iSCSI initiator-groups dynamic management</li> <li>• iSCSI bidirectional CHAP</li> </ul>
<b>Control</b> <ul style="list-style-type: none"> <li>• Storage and performance consumption</li> <li>• Monitoring</li> <li>• Volume Import</li> <li>• Cross Namespace Volume Access</li> </ul>	<b>Installation methods</b> <ul style="list-style-type: none"> <li>• Binary</li> <li>• Helm chart</li> <li>• Operator</li> <li>• GitOps</li> </ul>
<b>Choose your access mode</b> <ul style="list-style-type: none"> <li>• RWO (ReadWriteOnce, i.e 1↔1)</li> <li>• RWX (ReadWriteMany, i.e 1↔n)</li> <li>• ROX (ReadOnlyMany)</li> <li>• RWOP (ReadWriteOnce POD)</li> </ul>	<b>Choose your protocol</b> <ul style="list-style-type: none"> <li>• NFS</li> <li>• SMB</li> <li>• iSCSI</li> </ul>

Business critical container workloads need more than just persistent volumes. Their data management requirements require protection and migration of the application kubernetes objects as well.



Application data includes kubernetes objects in addition to the user data: Some examples are as follows:

- kubernetes objects such as pods specs, PVCs, deployments, services
- custom config objects such as config maps and secrets
- persistent data such as Snapshot copies, backups, clones
- custom resources such as CRs and CRDs

**NetApp Astra Control**, available as both fully-managed and self-managed software, provides orchestration for robust application data management. Refer to the [Astra documentation](#) for additional details on the Astra family of products.

This reference documentation provides validation of migration and protection of container-based applications, deployed on RedHat OpenShift container platform, using NetApp Astra Control Center. In addition, the solution provides high-level details for the deployment and the use of Red Hat Advanced Cluster Management (ACM) for managing the container platforms. The document also highlights the details for the integration of NetApp storage with Red Hat OpenShift container platforms using Astra Trident CSI provisioner. Astra Control Center is deployed on the hub cluster and is used to manage the container applications and their persistent storage lifecycle. Finally, it provides a solution for replication and failover and fail-back for container workloads on managed Red Hat OpenShift clusters in AWS (ROSA) using Amazon FSx for NetApp ONTAP (FSxN) as persistent storage.

### NetApp Solution with Managed Red Hat OpenShift Container platform workloads on AWS

Customers may be "born in the cloud" or may be at a point in their modernization journey when they are ready to move some select workloads or all workloads from their data centers to the cloud. They may choose to use provider-managed OpenShift containers and provider-managed NetApp storage in the cloud for running their workloads. They should plan and deploy the Managed Red Hat OpenShift container clusters (ROSA) in

the cloud for a successful production-ready environment for their container workloads. When they are in AWS cloud, they could also deploy FSx for NetApp ONTAP for the storage needs.

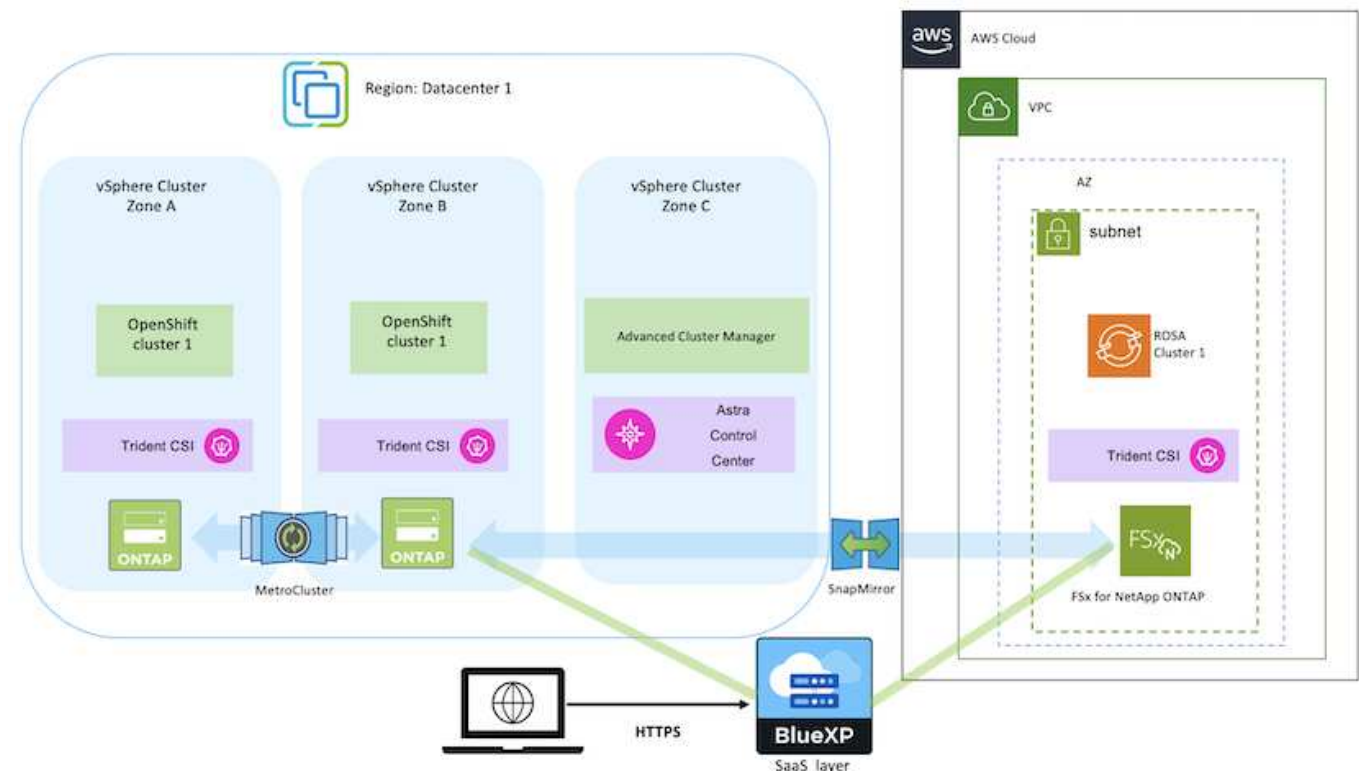
FSx for NetApp ONTAP delivers data protection, reliability, and flexibility for container deployments in AWS. Astra Trident serves as the dynamic storage provisioner to consume the persistent FSxN storage for customers' stateful applications.

As ROSA can be deployed in HA mode with control plane nodes spread across multiple availability zones, FSx ONTAP can also be provisioned with Multi-AZ option which provides high availability and protect against AZ failures.



There are no data transfer charges when accessing an Amazon FSx file system from the file system's preferred Availability Zone (AZ). For more info on pricing, refer [here](#).

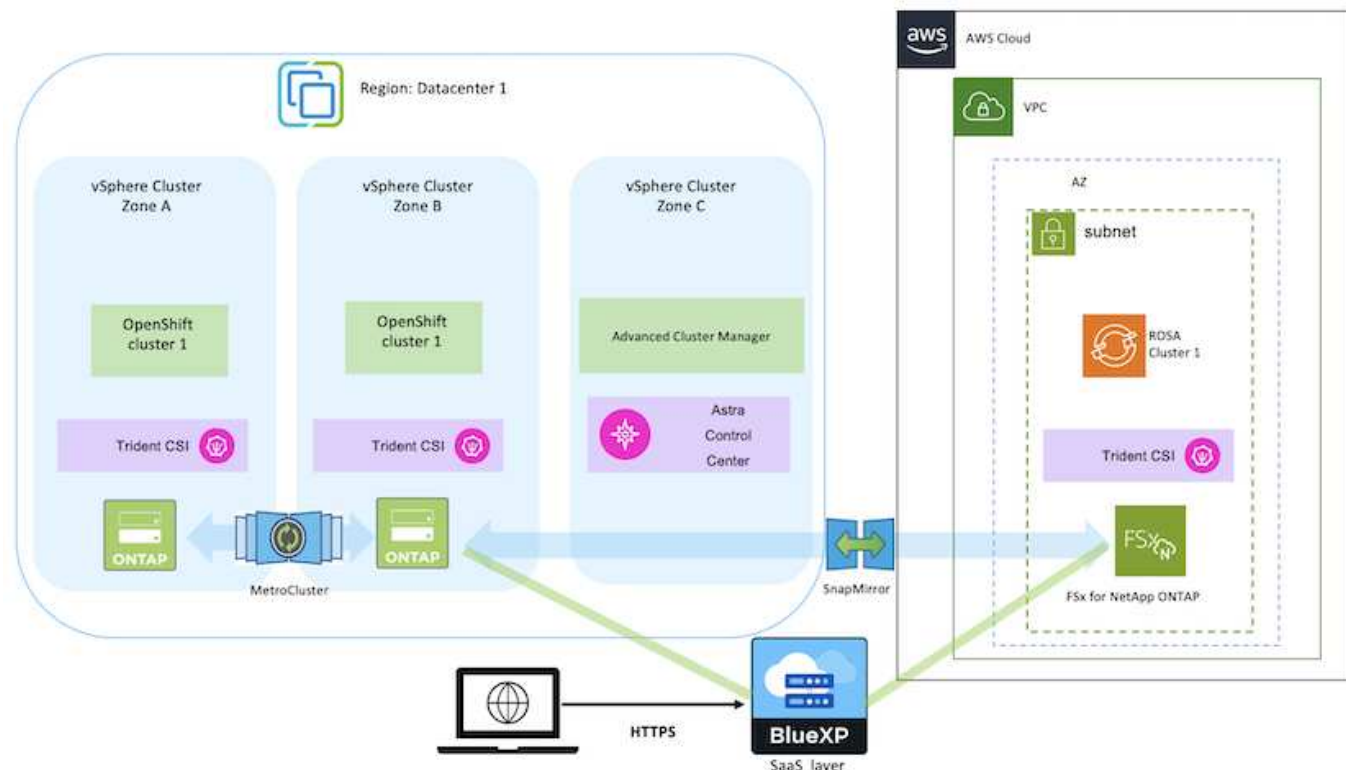
### Data protection and migration solution for OpenShift Container workloads



### Deploy and configure the Managed Red Hat OpenShift Container platform on AWS

This section describes a high-level workflow of setting up the Managed Red Hat OpenShift clusters on AWS(ROSA). It shows the use of Managed FSx for NetApp ONTAP (FSxN) as the storage backend by Astra Trident to provide persistent volumes. Details are provided about the deployment of FSxN on AWS using BlueXP. Also, details are provided about the use of BlueXP and OpenShift GitOps (Argo CD) to perform data protection and migration activities for the stateful applications on ROSA clusters.

Here is a diagram that depicts the ROSA clusters deployed on AWS and using FSxN as the backend storage.



This solution was verified by using two ROSA clusters in two VPCs in AWS. Each ROSA cluster was integrated with FSxN using Astra Trident. There are several ways of deploying ROSA clusters and FSxN in AWS. This high-level description of the setup provides documentation links for the specific method that was used. You can refer to the other methods in the relevant links provided in the [resources section](#).

The setup process can be broken down into the following steps:

### Install ROSA clusters

- Create two VPCs and set up VPC peering connectivity between the VPCs.
- Refer [here](#) for instructions to install ROSA clusters.

### Install FSxN

- Install FSxN on the VPCs from BlueXP.  
Refer [here](#) for BlueXP account creation and to get started.  
Refer [here](#) for installing FSxN.  
Refer [here](#) for creating a connector in AWS to manage the FSxN.
- Deploy FSxN using AWS.  
Refer [here](#) for deployment using AWS console.



## Install Trident on ROSA clusters (using Helm chart)

- Use Helm chart to install Trident on ROSA clusters.  
url for the Helm chart: <https://netapp.github.io/trident-helm-chart>

### Integration of FSxN with Astra Trident for ROSA clusters



OpenShift GitOps can be utilized to deploy Astra Trident CSI to all managed clusters as they get registered to ArgoCD using ApplicationSet.

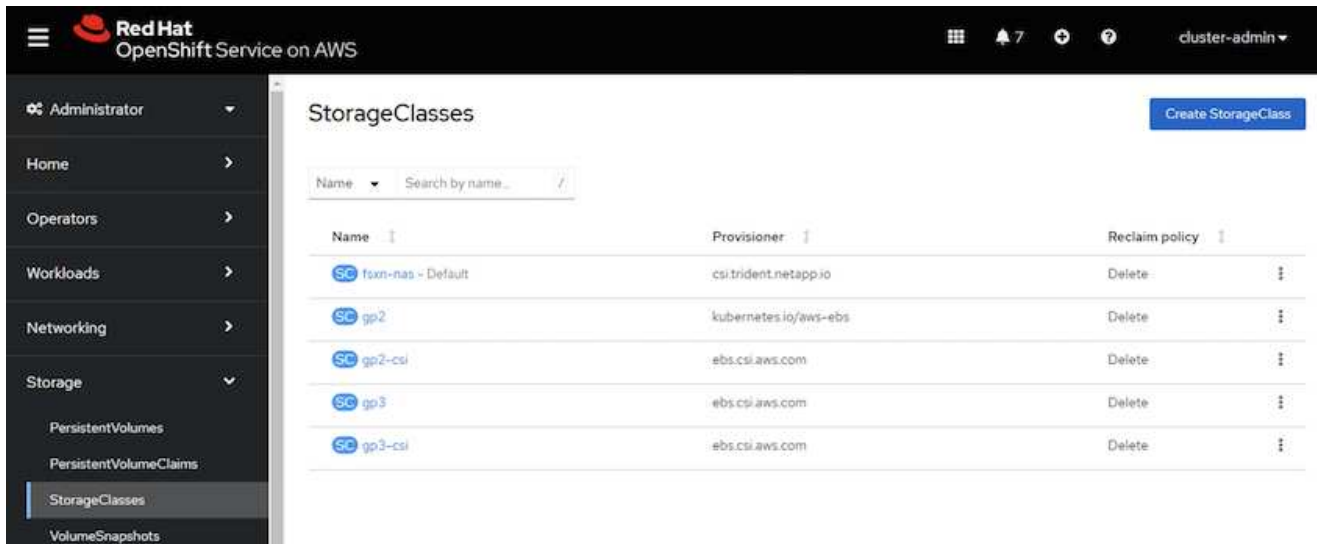
```
apiVersion: argoproj.io/v1alpha1
kind: ApplicationSet
metadata:
  name: trident-operator
spec:
  generators:
    - clusters: {}
      # selector:
      #   matchLabels:
      #     tridentversion: '23.04.0'
  template:
    metadata:
      name: '{{nameNormalized}}-trident'
    spec:
      destination:
        namespace: trident
        server: '{{server}}'
      source:
        repoURL: 'https://netapp.github.io/trident-helm-chart'
        targetRevision: 23.04.0
        chart: trident-operator
      project: default
      syncPolicy:
        syncOptions:
          - CreateNamespace=true
```





## Create backend and storage classes using Trident (for FSxN)

- Refer [here](#) for details about creating backend and storage class.
- Make the storage class created for FsxN with Trident CSI as default from OpenShift Console. See screenshot below:



## Deploy an application using OpenShift GitOps (Argo CD)

- Install OpenShift GitOps operator on the cluster. Refer to instructions [here](#).
- SetUp a new Argo CD instance for the cluster. Refer to instructions [here](#).

Open the console of Argo CD and deploy an app.

As an example, you can deploy a Jenkins App using Argo CD with a Helm Chart.

When creating the application, the following details were provided:

Project: default

cluster: <https://kubernetes.default.svc>

Namespace: Jenkins

The url for the Helm Chart: <https://charts.bitnami.com/bitnami>

Helm Parameters:

global.storageClass: fsxn-nas

## Data protection

This page shows the data protection options for Managed Red Hat OpenShift on AWS (ROSA) clusters using Astra Control Service. Astra Control Service (ACS) provides an easy-to-use graphical user-interface with which you can add clusters, define applications running on them, and perform application aware data management activities. ACS functions can also be accessed using an API that allows for automation of workflows.

Powering Astra Control (ACS or ACC) is NetApp Astra Trident. Astra Trident integrates several types of Kubernetes clusters such as Red Hat OpenShift, EKS, AKS, SUSE Rancher, Anthos etc., with various flavors of NetApp ONTAP storage such as FAS/AFF, ONTAP Select, CVO, Google Cloud Volumes Service, Azure

## NetApp Files and Amazon FSx for NetApp ONTAP.

This section provides details for the following data protection options using ACS:

- A video showing Backup and Restore of a ROSA application running in one region and restoring to another region.
- A video showing Snapshot and Restore of a ROSA application.
- Step-by-step details of installing a ROSA cluster, Amazon FSx for NetApp ONTAP, using NetApp Astra Trident to integrate with storage backend, installing a postgresql application on ROSA cluster, using ACS to create a snapshot of the application and restoring the application from it.
- A blog showing step-by-step details of creating and restoring from a snapshot for a mysql application on a ROSA cluster with FSx for ONTAP using ACS.

### Backup/Restore from Backup

The following video shows the backup of a ROSA application running in one region and restoring to another region.

[FSx NetApp ONTAP for Red Hat OpenShift Service on AWS](#)

### Snapshot/Restore from snapshot

The following video shows taking a snapshot of a ROSA application and restoring from the snapshot after.

[Snapshot/Restore for Applications on Red Hat OpenShift Service on AWS \(ROSA\)clusters with Amazon FSx for NetApp ONTAP storage](#)

### Blog

- [Using Astra Control Service for data management of apps on ROSA clusters with Amazon FSx storage](#)

### Step-by-Step Details to create snapshot and restore from it

#### Prerequisite setup

- [AWS account](#)
- [Red Hat OpenShift account](#)
- IAM user with [appropriate permissions](#) to create and access ROSA cluster
- [AWS CLI](#)
- [ROSA CLI](#)
- [OpenShift CLI\(oc\)](#)
- VPC with subnets and appropriate gateways and routes
- [ROSA Cluster installed](#) into the VPC
- [Amazon FSx for NetApp ONTAP](#) created in the same VPC
- Access to the ROSA cluster from [OpenShift Hybrid Cloud Console](#)

#### Next Steps

1. Create an admin user and login to the cluster.

2. Create a kubeconfig file for the cluster.
3. Install Astra Trident on the cluster.
4. Create a backend, storage class and snapshot class configuration using the Trident CSI provisioner.
5. Deploy a postgresql application on the cluster.
6. Create a database and add a record.
7. Add the cluster into ACS.
8. Define the application in ACS.
9. Create a snapshot using ACS.
10. Delete the database in the postgresql application.
11. Restore from a snapshot using ACS.
12. Verify your app has been restored from the snapshot.

## 1. Create an admin user and login to the cluster

Access the ROSA cluster by creating an admin user with the following command : (You need to create an admin user only if you did not create one at the time of installation)

```
rosa create admin --cluster=<cluster-name>
```

The command will provide an output that will look like the following. Login to the cluster using the `oc login` command provided in the output.

```
W: It is recommended to add an identity provider to login to this cluster.
See 'rosa create idp --help' for more information.
I: Admin account has been added to cluster 'my-rosa-cluster'. It may take up
to a minute for the account to become active.
I: To login, run the following command:
oc login https://api.my-rosa-cluster.abcd.p1.openshiftapps.com:6443 \
--username cluster-admin \
--password FWGYL-2mkJI-00000-00000
```



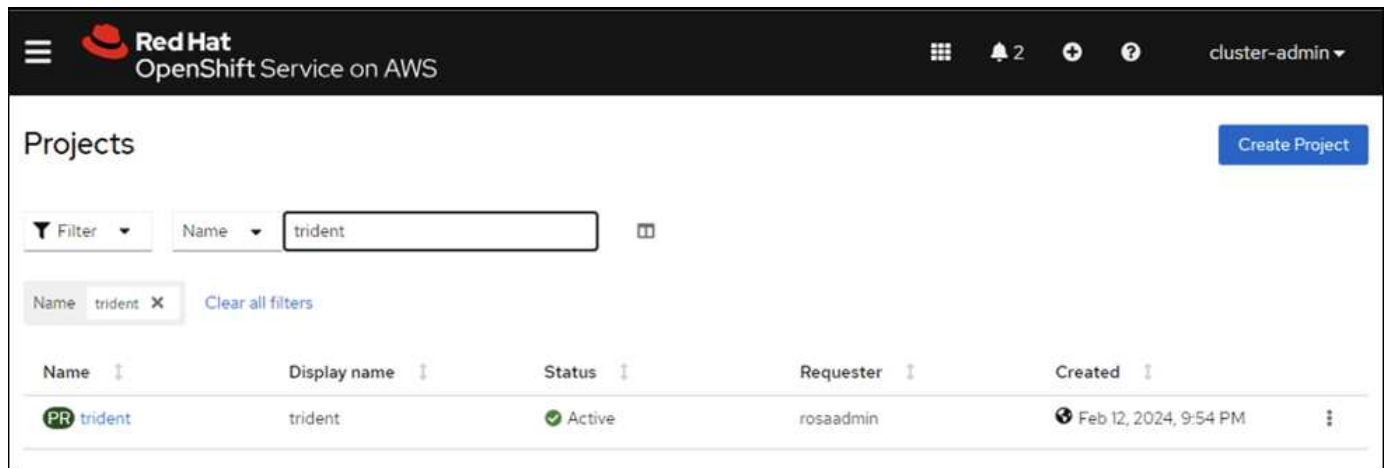
You can also login to the cluster using a token. If you already created an admin-user at the time of cluster creation, you can login to the cluster from the Red Hat OpenShift Hybrid Cloud console with the admin-user credentials. Then by clicking on the top right corner where it displays the name of the logged in user, you can obtain the `oc login` command (token login) for the command line.

## 2. Create a kubeconfig file for the cluster

Follow the procedures [here](#) to create a kubeconfig file for the ROSA cluster. This kubeconfig file will be used later when you add the cluster into ACS.

### 3. Install Astra Trident on the cluster

Install Astra Trident (latest version) on the ROSA cluster. To do this, you can follow any one of the procedures given [here](#). To install Trident using helm from the console of the cluster, first create a project called Trident.



Then from the Developer view, create a Helm chart repository. For the URL field use 'https://netapp.github.io/trident-helm-chart'. Then create a helm release for Trident operator.

## Create Helm Chart Repository

Add helm chart repository.

Configure via: ☒ Form view ☐ YAML view

### Scope type

☐ Namespaced scoped (ProjectHelmChartRepository)

Add Helm Chart Repository in the selected namespace.

☒ Cluster scoped (HelmChartRepository)

Add Helm Chart Repository at the cluster level and in all namespaces.

### Name \*

trident

A unique name for the Helm Chart repository.

### Display name

Astra Trident

A display name for the Helm Chart repository.

### Description

NetApp Astra Trident

A description for the Helm Chart repository.

☐ Disable usage of the repo in the developer catalog.

### URL \*

https://netapp.github.io/trident-helm-chart

Project: trident ▼

Developer Catalog > Helm Charts

# Helm Charts

Browse for charts that help manage complex installations and upgrades. Cluster administrators can customize the catalog. Alternatively, developers can [try to configure their own custom Helm Chart repository](#).

All items

CI/CD

Languages

Other

Chart Repositories

☒ Astra Trident (1)

☐ OpenShift Helm Charts (87)

Source

☐ Community (33)

☐ Partner (42)


☐ Red Hat (12)

All items

Q

Filter by keyword...

A-Z ▼



Helm Charts

Trident Operator

A Helm chart for deploying NetApp's Trident CSI storage provisioner using the Trident...

Verify all trident pods are running by going back to the Administrator view on the console and selecting pods in the trident project.

52

**Red Hat**  
 OpenShift Service on AWS

Administrator

Home

Operators

Workloads

Pods

Deployments

DeploymentConfigs

StatefulSets

Secrets

ConfigMaps

CronJobs

Jobs

DaemonSets

ReplicaSets

ReplicationControllers

HorizontalPodAutoscalers

PodDisruptionBudgets

Networking

Project: trident

Pods

Filter Name Search by name...

Name	Status	Ready	Restarts	Owner	Memory
trident-controller-69cff44ddf-4dqnj	Running	6/6	0	trident-controller-69cff44ddf	-
trident-node-linux-4b6fm	Running	2/2	0	trident-node-linux	-
trident-node-linux-4sckw	Running	2/2	0	trident-node-linux	-
trident-node-linux-7l42w	Running	2/2	0	trident-node-linux	-
trident-node-linux-dbhp4	Running	2/2	0	trident-node-linux	-
trident-node-linux-gj5km	Running	2/2	0	trident-node-linux	-
trident-node-linux-r79c8	Running	2/2	0	trident-node-linux	-
trident-node-linux-tzwdp	Running	2/2	0	trident-node-linux	-
trident-node-linux-vdvxt	Running	2/2	0	trident-node-linux	-
trident-operator-7f7fd45c68-6crqb	Running	1/1	0	trident-operator-7f7fd45c68	-

#### 4. Create a backend, storage class and snapshot class configuration using the Trident CSI provisioner

Use the yaml files shown below to create a trident backend object, storage class object and the Volumesnapshot object. Be sure to provide the credentials to your Amazon FSx for NetApp ONTAP file system you created, the management LIF and the vserver name of your file system in the configuration yaml for the backend. To get those details, go to the AWS console for Amazon FSx and select the file system, navigate to the Administration tab. Also, click on update to set the password for the `fsxadmin` user.



You can use the command line to create the objects or create them with the yaml files from the hybrid cloud console.

FSx > File systems > fs-049f9a23aac951429

## fsx-for-rosa (fs-049f9a23aac951429)

▼ Summary

File system ID fs-049f9a23aac951429	SSD storage capacity 1024 GiB	<button>Update</button>	Availability Zones us-west-2b
Lifecycle state Available	Throughput capacity 128 MB/s	<button>Update</button>	Creation time 2024-02-12T20:15:23-05:00
File system type ONTAP	Provisioned IOPS 3072	<button>Update</button>	
Deployment type Single-AZ	Number of HA pairs 1		

Network & security | Monitoring & performance | **Administration** | Storage virtual machines | Volumes | Backups | Updates | Tags

### ONTAP administration

Management endpoint - DNS name management.fs-049f9a23aac951429.fsx.us-west-2.amazonaws.com	Management endpoint - IP address 10.49.9.135	ONTAP administrator username fsxadmin
Inter-cluster endpoint - DNS name intercluster.fs-049f9a23aac951429.fsx.us-west-2.amazonaws.com	Inter-cluster endpoint - IP address 10.49.9.49	ONTAP administrator password <button>Update</button>
	10.49.9.251	

## Trident Backend Configuration

```

apiVersion: v1
kind: Secret
metadata:
  name: backend-tbc-ontap-nas-secret
type: Opaque
stringData:
  username: fsxadmin
  password: <password>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: ontap-nas
spec:
  version: 1
  storageDriverName: ontap-nas
  managementLIF: <management lif>
  backendName: ontap-nas
  svm: fsx
  credentials:
    name: backend-tbc-ontap-nas-secret

```

## Storage Class



```

apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: ontap-nas
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-nas"
  media: "ssd"
  provisioningType: "thin"
  snapshots: "true"
allowVolumeExpansion: true

```

## snapshot class

```

apiVersion: snapshot.storage.k8s.io/v1
kind: VolumeSnapshotClass
metadata:
  name: trident-snapshotclass
driver: csi.trident.netapp.io
deletionPolicy: Delete

```

Verify that the backend, storage class and the trident-snapshotclass objects are created by issuing the commands shown below.

```

[ec2-user@ip-10-49-11-132 storage]$ kubectl get tbc -n trident
NAME          BACKEND NAME    BACKEND UUID          PHASE    STATUS
ontap-nas     ontap-nas       8a5e4583-2dac-46bb-b01e-fa7c3816f121  Bound    Success
[ec2-user@ip-10-49-11-132 storage]$ kubectl get sc
NAME          PROVISIONER          RECLAIMPOLICY    VOLUMEBINDINGMODE    ALLOWVOLUMEEXPANSION    AGE
gp2           kubernetes.io/aws-ebs Delete            WaitForFirstConsumer  true                    3h23m
gp2-csi       ebs.csi.aws.com      Delete            WaitForFirstConsumer  true                    3h19m
gp3 (default) ebs.csi.aws.com      Delete            WaitForFirstConsumer  true                    3h23m
gp3-csi       ebs.csi.aws.com      Delete            WaitForFirstConsumer  true                    3h19m
ontap-nas     csi.trident.netapp.io Delete            Immediate              true                    141m
[ec2-user@ip-10-49-11-132 storage]$ kubectl get Volumesnapshotclass
NAME          DRIVER          DELETIONPOLICY    AGE
csi-aws-vsc    ebs.csi.aws.com Delete            3h19m
trident-snapshotclass csi.trident.netapp.io Delete            6m56s
[ec2-user@ip-10-49-11-132 storage]$

```

At this time, an important modification you need to make is to set ontap-nas as the default storage class instead of gp3 so that the postgresql app you deploy later can use the default storage class. In the Openshift console of your cluster, under Storage select StorageClasses. Edit the annotation of the current default class to be false and add the annotation storageclass.kubernetes.io/is-default-class set to true for the ontap-nas storage class.

**Edit annotations**

Key: storageclass.kubernetes.io/is-... Value: false

+ Add more

Cancel Save

Name	Provisioner	Reclaim policy
SC gp2	kubernetes.io/aws-ebs	Delete
SC gp2-csi	ebs.csi.aws.com	Delete
SC gp3 - Default	ebs.csi.aws.com	Delete
SC gp3-csi	ebs.csi.aws.com	Delete
SC ontap-nas	csitrident.netapp.io	Delete

**StorageClasses** Create StorageClass

Name Search by name...

Name	Provisioner	Reclaim policy
SC gp2	kubernetes.io/aws-ebs	Delete
SC gp2-csi	ebs.csi.aws.com	Delete
SC gp3	ebs.csi.aws.com	Delete
SC gp3-csi	ebs.csi.aws.com	Delete
SC ontap-nas - Default	csitrident.netapp.io	Delete

## 5. Deploy a postgresql application on the cluster

You can deploy the application from the command line as follows:

```
helm install postgresql bitnami/postgresql -n postgresql --create-namespace
```

```
[ec2-user@ip-10-49-11-132 astra]$ helm install postgresql bitnami/postgresql -n postgresql --create-namespace
NAME: postgresql
LAST DEPLOYED: Tue Feb 13 14:46:16 2024
NAMESPACE: postgresql
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
CHART NAME: postgresql
CHART VERSION: 14.0.4
APP VERSION: 16.2.0

** Please be patient while the chart is being deployed **

PostgreSQL can be accessed via port 5432 on the following DNS names from within your cluster:

    postgresql.postgresql.svc.cluster.local - Read/Write connection

To get the password for "postgres" run:

    export POSTGRES_PASSWORD=$(kubectl get secret --namespace postgresql postgresql -o jsonpath="{.data.postgres-password}" | base64 -d)

To connect to your database run the following command:

    kubectl run postgresql-client --rm --tty -i --restart='Never' --namespace postgresql --image docker.io/bitnami/postgresql:16.2.0-debian-11-r1 --env="PGPASSWORD=$POSTGRES_PASSWORD" \
    --command -- psql --host postgresql -U postgres -d postgres -p 5432

> NOTE: If you access the container using bash, make sure that you execute "/opt/bitnami/scripts/postgresql/entrypoint.sh /bin/bash" in order to avoid
the error "psql: local user with ID 1001 does not exist"

To connect to your database from outside the cluster execute the following commands:

    kubectl port-forward --namespace postgresql svc/postgresql 5432:5432 &
    PGPASSWORD=$POSTGRES_PASSWORD psql --host 127.0.0.1 -U postgres -d postgres -p 5432

WARNING: The configured password will be ignored on new installation in case when previous PostgreSQL release was deleted through the helm command. In that
case, old PVC will have an old password, and setting it through helm won't take effect. Deleting persistent volumes (PVs) will solve the issue.
[ec2-user@ip-10-49-11-132 astra]$
```

If you do not see the application pods running, then there might be an error caused due to security context constraints.

```
[ec2-user@ip-10-49-11-132 astra]$ kubectl get all -n postgresql
NAME                                TYPE          CLUSTER-IP    EXTERNAL-IP    PORT(S)    AGE
service/postgresql                  ClusterIP     172.30.245.50  <none>         5432/TCP   12m
service/postgresql-hl               ClusterIP     None           <none>         5432/TCP   12m

NAME                                READY    AGE
statefulset.apps/postgresql         0/1      12m
[ec2-user@ip-10-49-11-132 astra]$ kubectl get events -n postgresql
LAST SEEN   TYPE      REASON              OBJECT                                          MESSAGE
12m39s      Normal    WaitForFirstConsumer persistentvolumeclaim/data-postgresql-0       waiting for first consumer to be created before binding
12m          Normal    SuccessfulCreate     statefulset/postgresql                        create Claim data-postgresql-0 Pod postgresql-0 in StatefulSet postg
resql success
107s        Warning   FailedCreate         statefulset/postgresql                        create Pod postgresql-0 in StatefulSet postgresql failed error: pods
"postgresql-0" is forbidden: unable to validate against any security context constraint: [provider "trident-controller": Forbidden: not usable by user or
serviceaccount, provider "anyuid": Forbidden: not usable by user or serviceaccount, provider "restricted-v2": .spec.securityContext.fsGroup: Invalid value: [
1001]: 1001 is not an allowed group, provider "restricted-v2": .containers[0].runAsUser: Invalid value: 1001: must be in the ranges: [1001010000, 1001
019999], provider "restricted": Forbidden: not usable by user or serviceaccount, provider "nonroot-v2": Forbidden: not usable by user or serviceaccount, pr
ovider "nonroot": Forbidden: not usable by user or serviceaccount, provider "pcap-dedicated-admins": Forbidden: not usable by user or serviceaccount, provi
der "hostmount-anyuid": Forbidden: not usable by user or serviceaccount, provider "machine-api-termination-handler": Forbidden: not usable by user or servi
ceaccount, provider "hostnetwork-v2": Forbidden: not usable by user or serviceaccount, provider "hostnetwork": Forbidden: not usable by user or serviceacco
unt, provider "hostaccess": Forbidden: not usable by user or serviceaccount, provider "splunkforwarder": Forbidden: not usable by user or serviceaccount, p
rovider "trident-node-linux": Forbidden: not usable by user or serviceaccount, provider "node-exporter": Forbidden: not usable by user or serviceaccount, p
rovider "privileged": Forbidden: not usable by user or serviceaccount]
[ec2-user@ip-10-49-11-132 astra]$
```

Fix the error by editing the `runAsUser` and `fsGroup` fields in `statefulset.apps/postgresql` object with the uid that is in the output of the `oc get project` command as shown below.

```
[ec2-user@ip-10-49-11-132 astra]$ oc get project postgresql -o yaml | grep uid-range
openshift.io/sa.scc.uid-range: 1001010000/10000
[ec2-user@ip-10-49-11-132 astra]$ oc edit -n postgresql statefulset.apps/postgresql
statefulset.apps/postgresql edited
[ec2-user@ip-10-49-11-132 astra]$
```

postgresql app should be running and using persistent volumes backed by Amazon FSx for NetApp ONTAP storage.

```
[ec2-user@ip-10-49-11-132 astra]$ oc get pods -n postgresql
```

NAME	READY	STATUS	RESTARTS	AGE
postgresql-0	1/1	Running	0	2m46s

```
[ec2-user@ip-10-49-11-132 astra]$
```

```
[ec2-user@ip-10-49-11-132 storage]$ kubectl get pvc -n postgresql
```

NAME	STATUS	VOLUME	CAPACITY	ACCESS MODES	STORAGECLASS	AGE
data-postgresql-0	Bound	pvc-dd09524a-de75-4825-9424-03a9b91195ca	8Gi	RWO	ontap-nas	4m2s

```
[ec2-user@ip-10-49-11-132 storage]$
```

## 6. Create a database and add a record

```
[ec2-user@ip-10-49-11-132 astra]$ export POSTGRES_PASSWORD=$(kubectl get secret --namespace postgresql postgresql -o jsonpath="{.data.postgres-password}" | base64 -d)
[ec2-user@ip-10-49-11-132 astra]$ kubectl run postgresql-client --rm --tty -i --restart='Never' --namespace postgresql --image docker.io/bitnami/postgresql:16.2.0-debian-11-r1 --env="PGPASSWORD=$POSTGRES_PASSWORD" \
> --command -- psql --host postgresql -U postgres -d postgres -p 5432
Warning: would violate PodSecurity "restricted:vi1.24": allowPrivilegeEscalation != false (container "postgresql-client" must set securityContext.allowPrivilegeEscalation=false), unrestricted capabilities (container "postgresql-client" must set securityContext.capabilities.drop=["ALL"]), runAsNonRoot != true (pod or container "postgresql-client" must set securityContext.runAsNonRoot=true), seccompProfile (pod or container "postgresql-client" must set securityContext.seccompProfile.type to "RuntimeDefault" or "Localhost")
If you don't see a command prompt, try pressing enter.

postgres=# CREATE DATABASE erp;
CREATE DATABASE
postgres=# \c erp
You are now connected to database "erp" as user "postgres".
erp=# CREATE TABLE PERSONS(ID INT PRIMARY KEY NOT NULL, FIRSTNAME TEXT NOT NULL, LASTNAME TEXT NOT NULL);
CREATE TABLE
erp=# INSERT INTO PERSONS VALUES(1,'John','Doe');
INSERT 0 1
erp=# \dt
          List of relations
Schema | Name   | Type  | Owner
-----+-----+-----+-----
public | persons | table | postgres
(1 row)

erp=# SELECT * FROM persons;
 id | firstame | lastname
-----+-----+-----
  1 | John    | Doe
(1 row)
```

## 7. Add the cluster into ACS

Log in to ACS. Select cluster and click on Add. Select other and upload or paste the kubeconfig file.



**Add cluster**

STEP 1/3: DETAILS

PROVIDER

Microsoft Azure

Google Cloud Platform

Amazon Web Services

Other

KUBECONFIG

Please ensure that the kubeconfig used for this cluster has a long-lived token associated with it.

Provide Astra Control access to your Kubernetes clusters by entering a kubeconfig credential. Follow these [instructions](#) on how to create a dedicated admin-role kubeconfig.

Upload file

Paste or type

```

XJu2XR1cy5pby9sZXU2aWN1YWNjb3VudC9sZXU2aWN1LWFjY291bnQubmFtZSI6ImFzdHJhY29udHJvbmC1z2XU2aWN1LWFjY291bnQ1LCJrdWJ1cm5ldGZlLmlvL3N1cnZpY2VhY2NvdW50L3N1cnZpY2U0YWNjb3VudC51aWQ1OiI4NzFhOTI4MC0wMTBjLTBmYzAtOWFkNS0zZDI5NzA2N2N1NToiLCJzdWIiOiJzeXN0ZW06c2VydmljZWVjY291bnQ6ZGVmYXVsdDphc3RyYWNvbnRyb2wtc2VydmljZS1hY2NvdW50In0.M7-IRxoaK0e7S-LkW-8ZDY0ShQ5Uo1a5bJ-0SId5rOEbvfcQ3tSf40VC72nM4BqYbN8cm0y0V8IpF3OG7cYA9XAI dwX98xAXJ00TZUOG2xbyLWfOqLCFDk3_uS9uqU63t8LLmeenCBi0m9PaD3XWHFZ2cTXpdKqtzWfmbLxYhuN1CzBMY7S55MvNB2WD_eikptN02a1vaWmIZjrUQL0_q8Uj2Exe9vVH1KPkf0CxU4TvHncbathvL6mZ1N7Om

```

Cancel

Next →

Click **Next** and select `ontap-nas` as the default storage class for ACS. Click **Next**, review the details and **Add** the cluster.

**Add cluster**

STEP 2/3: STORAGE

STORAGE

☒
Assign a new default storage class

The following storage classes are available on the cluster.

Set default	Storage class	Storage provisioner	Reclaim policy	Binding mode	Eligibility
<input type="radio"/>	gp2	kubernetes.io/aws-ebs	Delete	WaitForFirstConsumer	Ineligible
<input type="radio"/>	gp2-csi	ebs.csi.aws.com	Delete	WaitForFirstConsumer	Eligible
<input type="radio"/>	gp3	ebs.csi.aws.com	Delete	WaitForFirstConsumer	Eligible
<input type="radio"/>	gp3-csi	ebs.csi.aws.com	Delete	WaitForFirstConsumer	Eligible
<input checked="" type="radio"/>	ontap-nas <small>Default</small>	csi.trident.netapp.io	Delete	Immediate	Eligible

← Back

Next →

## 8. Define the application in ACS

Define the postgresql application in ACS. From the landing page, select **Applications**, **Define** and fill in the appropriate details. Click **Next** a couple of times, Review the details and click **Define**. The application gets

added to ACS.

Add cluster

STEP 2/3: STORAGE

STORAGE

✓

Assign a new default storage class

The following storage classes are available on the cluster.

Set default	Storage class	Storage provisioner	Reclaim policy	Binding mode	Eligibility
<input type="radio"/>	gp2	kubernetes.io/aws-ebs	Delete	WaitForFirstConsumer	Ineligible
<input type="radio"/>	gp2-csi	ebs.csi.aws.com	Delete	WaitForFirstConsumer	Eligible
<input type="radio"/>	gp3	ebs.csi.aws.com	Delete	WaitForFirstConsumer	Eligible
<input type="radio"/>	gp3-csi	ebs.csi.aws.com	Delete	WaitForFirstConsumer	Eligible
<input checked="" type="radio"/>	ontap-nas <span>Default</span>	csi.trident.netapp.io	Delete	Immediate	Eligible

← Back

Next →

9. Create a snapshot using ACS

There are many ways to create a snapshot in ACS. You can select the application and create a snapshot from the page that shows the details of the application. You can click on Create snapshot to create an on-demand snapshot or configure a protection policy.

Create an on-demand snapshot by simply clicking on **Create snapshot**, providing a name, reviewing the details, and clicking on **Snapshot**. The snapshot state changes to Healthy after the operation is completed.

Dashboard

Applications

Clusters

Cloud instances

Buckets

Account

Activity

Support

Data protection

Storage

Resources

Execution hooks

Activity

Tasks

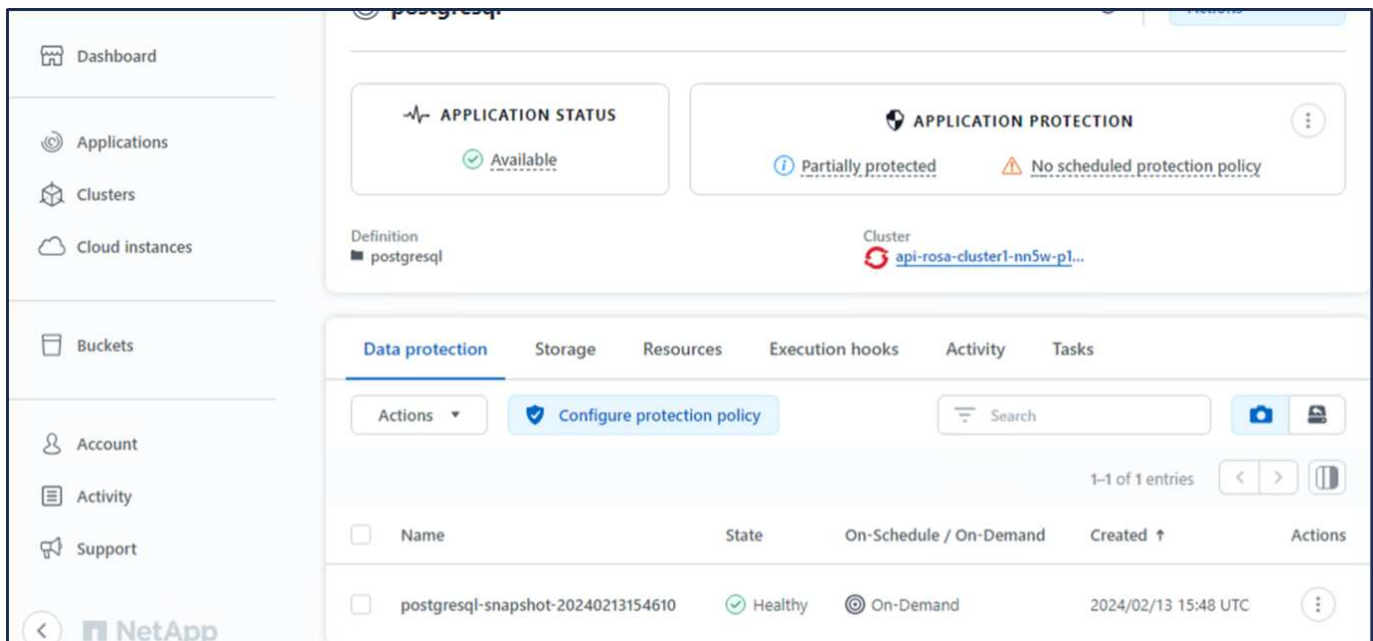
Actions

Configure protection policy

Search

0-0 of 0 entries

<input type="checkbox"/>	Name	State	On-Schedule / On-Demand	Created ↑	Actions
<div><div></div><div>You don't have any snapshots</div><div>After you have created a snapshot, it will be listed here</div><div>Create snapshot</div></div>					



## 10. Delete the database in the postgresql application

Log back into postgresql, list the available databases, delete the one you created previously and list again to ensure that the database has been deleted.

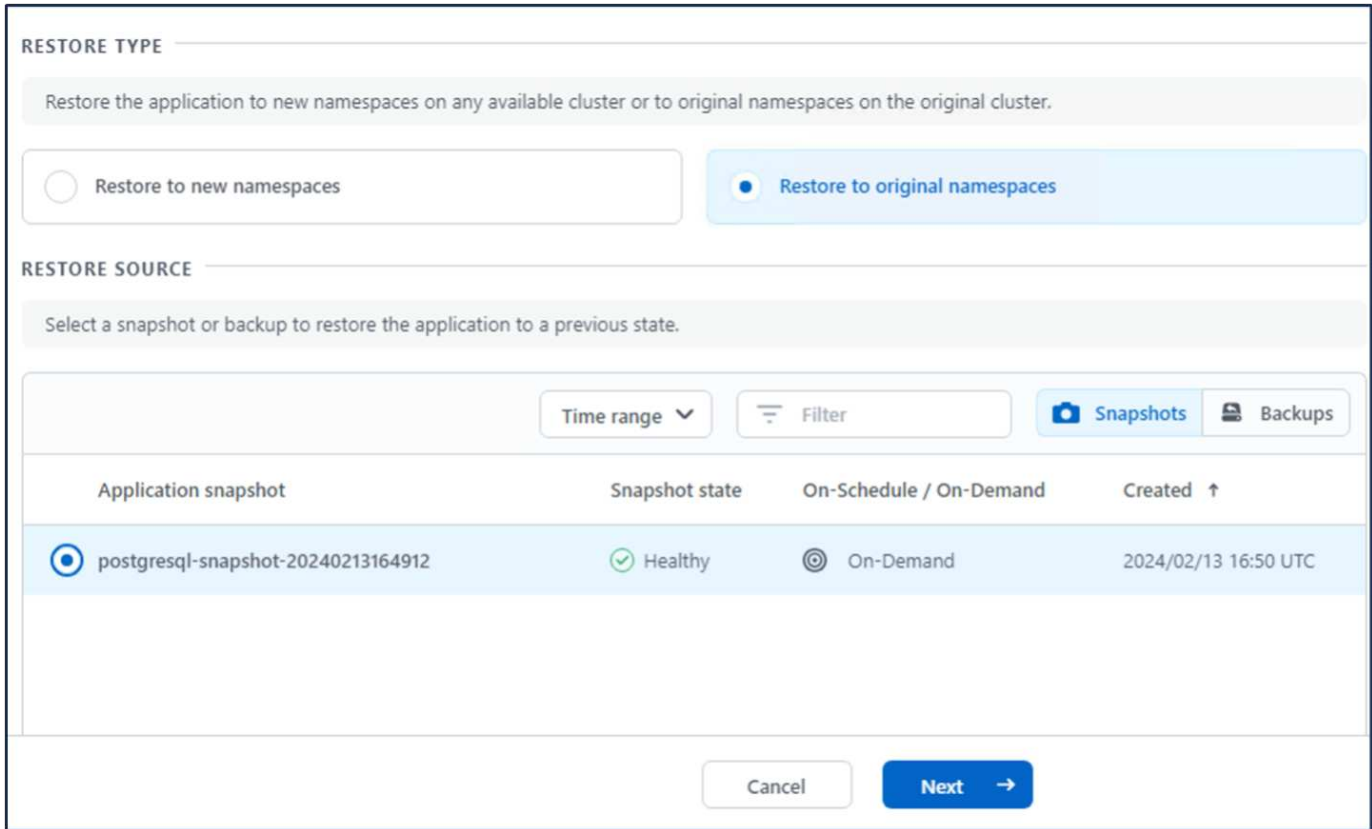
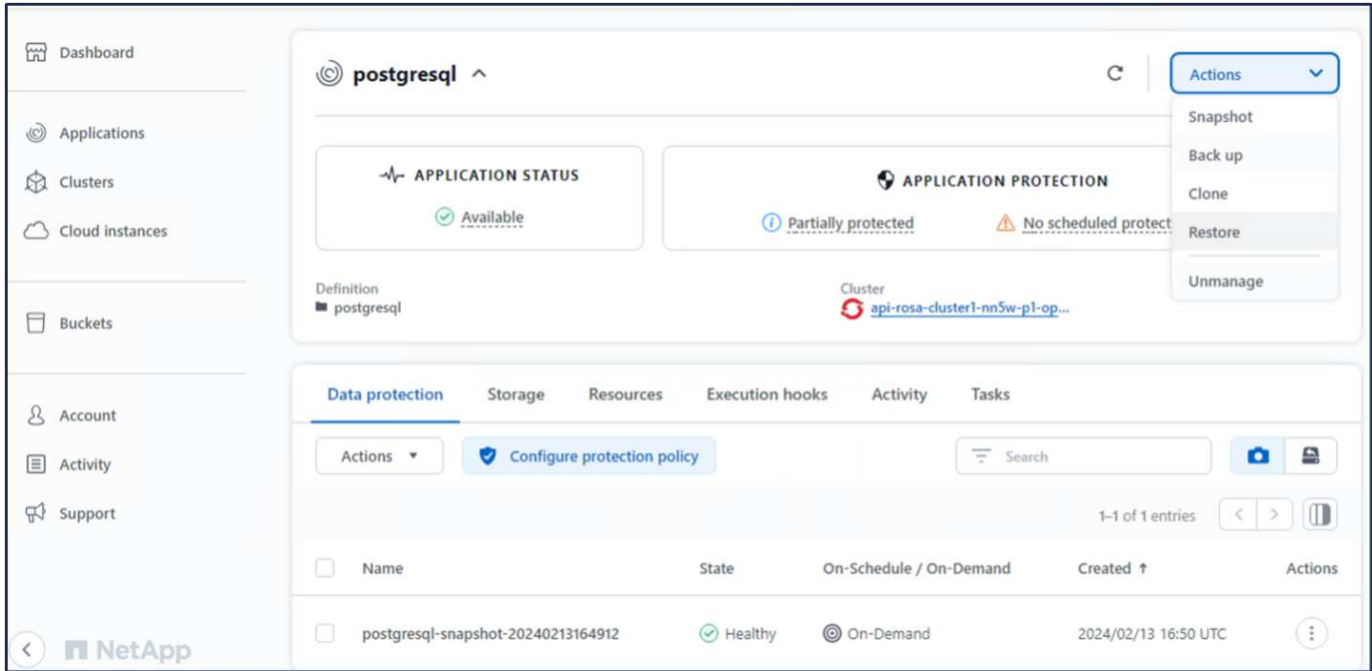
```
postgres=# \l
      List of databases
  Name | Owner | Encoding | Locale Provider | Collate | Ctype | ICU Locale | ICU Rules | Access privileges
-----+-----+-----+-----+-----+-----+-----+-----+-----
erp    | postgres | UTF8 | libc | en_US.UTF-8 | en_US.UTF-8 | | | =c/postgres
postgres | postgres | UTF8 | libc | en_US.UTF-8 | en_US.UTF-8 | | | =c/postgres
template0 | postgres | UTF8 | libc | en_US.UTF-8 | en_US.UTF-8 | | | =c/postgres
template1 | postgres | UTF8 | libc | en_US.UTF-8 | en_US.UTF-8 | | | =c/postgres
(4 rows)

postgres=# DROP DATABASE erp;
DROP DATABASE
postgres=# \l
      List of databases
  Name | Owner | Encoding | Locale Provider | Collate | Ctype | ICU Locale | ICU Rules | Access privileges
-----+-----+-----+-----+-----+-----+-----+-----+-----
postgres | postgres | UTF8 | libc | en_US.UTF-8 | en_US.UTF-8 | | | =c/postgres
template0 | postgres | UTF8 | libc | en_US.UTF-8 | en_US.UTF-8 | | | =c/postgres
template1 | postgres | UTF8 | libc | en_US.UTF-8 | en_US.UTF-8 | | | =c/postgres
(3 rows)
```

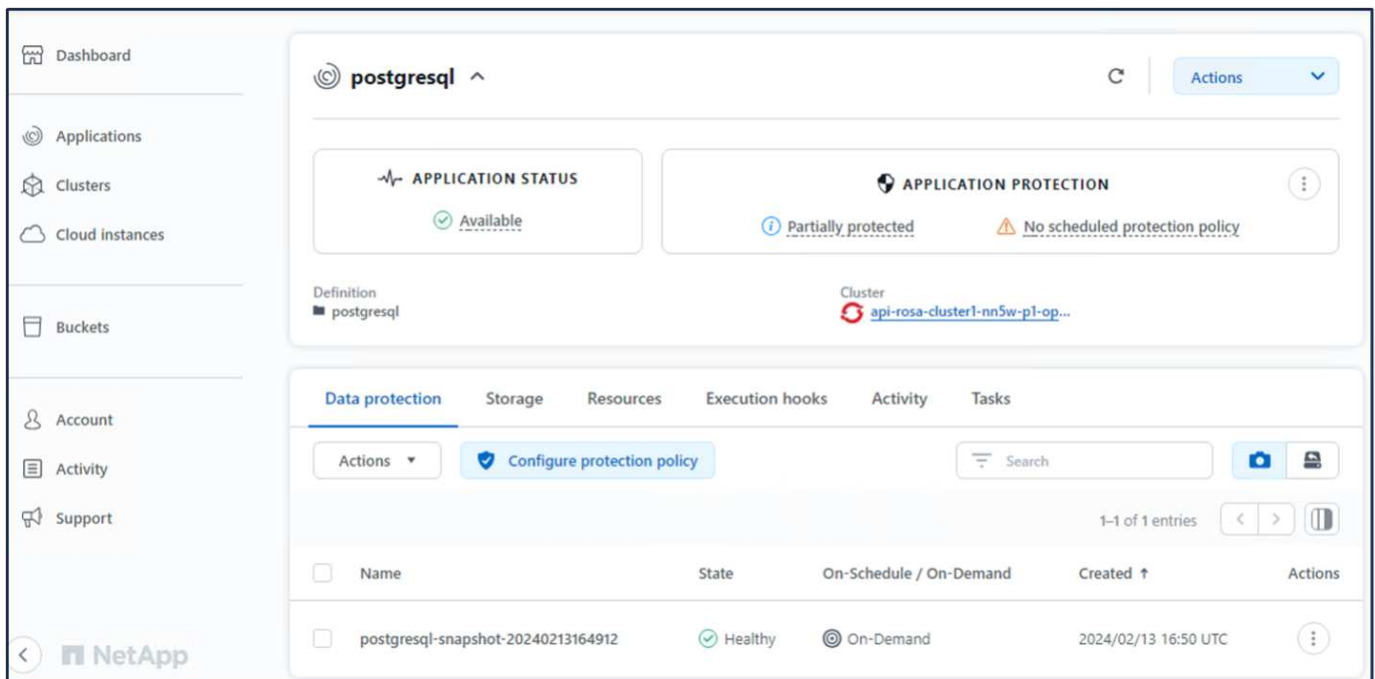
## 11. Restore from a snapshot using ACS

To restore the application from a snapshot, go to ACS UI landing page, select the application and select

Restore. You need to pick a snapshot or a backup from which to restore. (Typically, you would have multiple created based on a policy that you have configured). Make appropriate choices in the next couple of screens and then click on **Restore**. The application status moves from Restoring to Available after it has been restored from the snapshot.







## 12. Verify your app has been restored from the snapshot

Login to the postgresql client and you should now see the table and the record in the table that you previously had. That's it. Just by clicking a button, your application has been restored to a previous state. That is how easy we make it for our customers with Astra Control.

```
[ec2-user@ip-10-49-11-132 ~]$ kubectl run postgresql-client --rm --tty -i --restart='Never' --namespace postgresql --image docker.io/bitnami/postgresql:16.2.0-debian-11-r1 --env="PGPASSWORD=$POSTGRES_PASSWORD" --command -- psql --host postgresql -U postgres -d postgres -p 5432
Warning: would violate PodSecurity "restricted:vl.24": allowPrivilegeEscalation != false (container "postgresql-client" must set securityContext.allowPrivilegeEscalation=false), unrestricted capabilities (container "postgresql-client" must set securityContext.capabilities.drop=["ALL"]), runAsNonRoot != true (pod or container "postgresql-client" must set securityContext.runAsNonRoot=true), seccompProfile (pod or container "postgresql-client" must set securityContext.seccompProfile.type to "RuntimeDefault" or "Localhost")
If you don't see a command prompt, try pressing enter.

postgres=# \l
          List of databases
  Name | Owner  | Encoding | Locale Provider | Collate | Ctype  | ICU Locale | ICU Rules | Access privileges
-----+-----+-----+-----+-----+-----+-----+-----+-----
erp    | postgres | UTF8     | libc             | en_US.UTF-8 | en_US.UTF-8 |              |           |
postgres | postgres | UTF8     | libc             | en_US.UTF-8 | en_US.UTF-8 |              |           |
template0 | postgres | UTF8     | libc             | en_US.UTF-8 | en_US.UTF-8 |              |           |
template1 | postgres | UTF8     | libc             | en_US.UTF-8 | en_US.UTF-8 |              |           |
(4 rows)

postgres=# \c erp
You are now connected to database "erp" as user "postgres".
erp=# \dt
          List of relations
 Schema | Name  | Type  | Owner
-----+-----+-----+-----
 public | persons | table | postgres
(1 row)

erp=# SELECT * from PERSONS;
 id | firstame | lastname
----+-----+-----
  1 | John    | Doe
(1 row)
```

## Data migration

This page shows the data migration options for container workloads on Managed Red Hat OpenShift clusters using FSx for NetApp ONTAP for persistent storage.

## Data Migration

Red Hat OpenShift service on AWS as well as FSx for NetApp ONTAP (FSxN) are part of their service portfolio by AWS. FSxN is available on Single AZ or Multi-AZ options.

Multi-Az option provides data protection from availability zone failure.

FSxN can be integrated with Astra Trident to provide persistent storage for applications on ROSA clusters.

### Integration of FSxN with Trident using Helm chart

#### [ROSA Cluster Integration with Amazon FSx for ONTAP](#)

The migration of container applications involves:

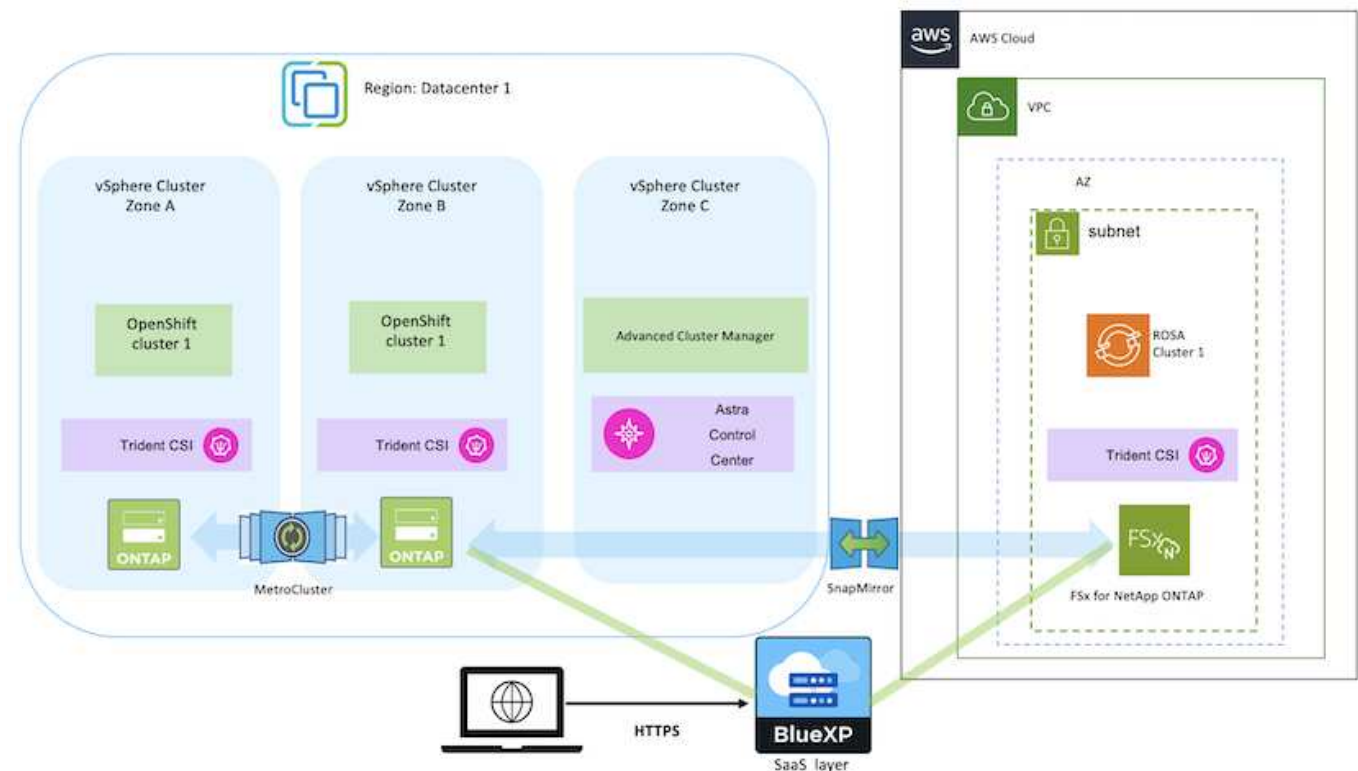
- Persistent volumes: this can be accomplished using BlueXP.  
Another option is to use Astra Control Center to handle container application migrations from on-premises to the cloud environment. Automation can be used for the same purpose.
- Application metadata: this can be accomplished using OpenShift GitOps (Argo CD).

### Failover and Fail-back of applications on ROSA cluster using FSxN for persistent storage

The following video is a demonstration of application failover and fail-back scenarios using BlueXP and Argo CD.

#### [Failover and Fail-back of applications on ROSA cluster](#)

### Data protection and migration solution for OpenShift Container workloads



# Data protection for Container Apps in OpenShift Container Platform using OpenShift API for Data Protection (OADP)

Author: Banu Sundhar, NetApp

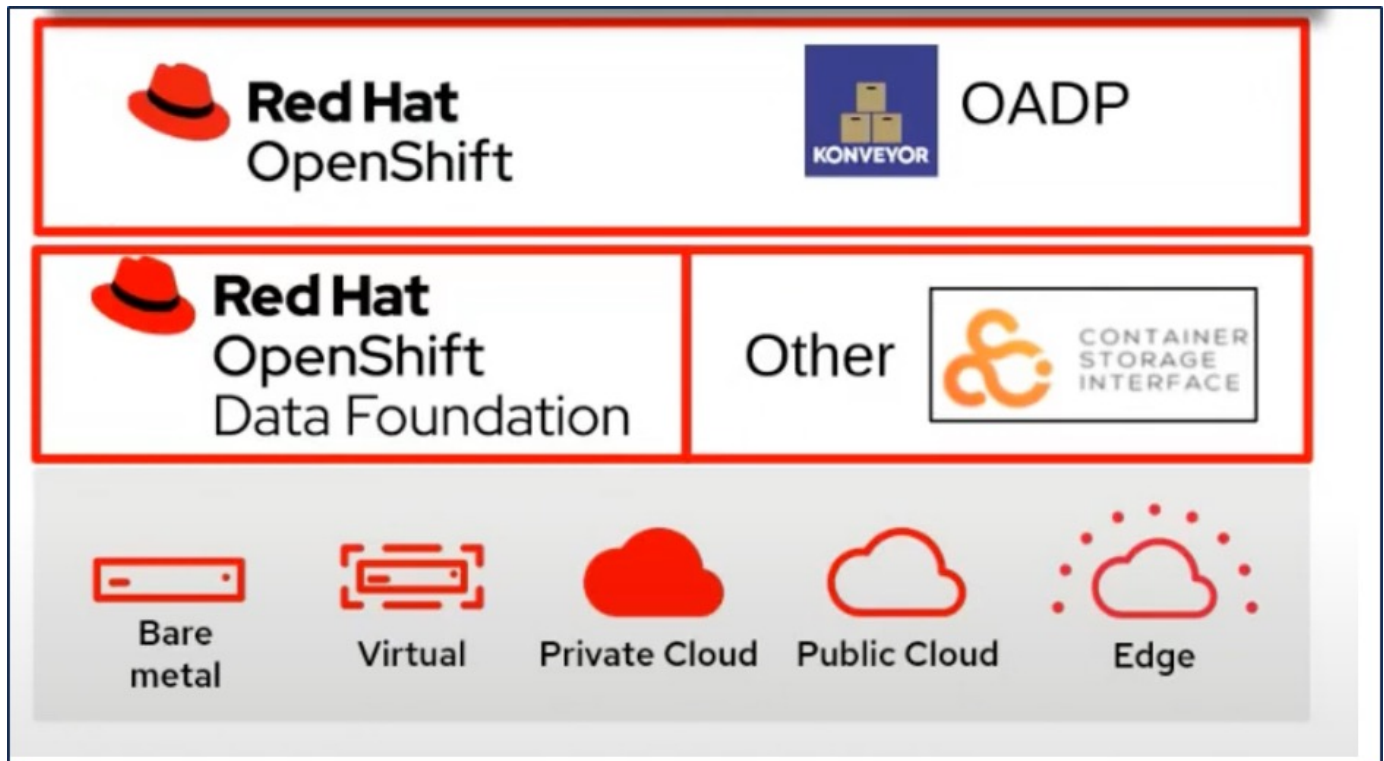
This section of the reference document provides details for creating backups of Container Apps using the OpenShift API for Data Protection (OADP) with Velero on NetApp ONTAP S3 or NetApp StorageGRID S3. The backups of namespace scoped resources including Persistent Volumes(PVs) of the app are created using CSI Astra Trident Snapshots.

The persistent storage for container apps can be backed by ONTAP storage integrated to the OpenShift Cluster using [Astra Trident CSI](#). In this section we use [OpenShift API for Data Protection \(OADP\)](#) to perform backup of apps including its data volumes to

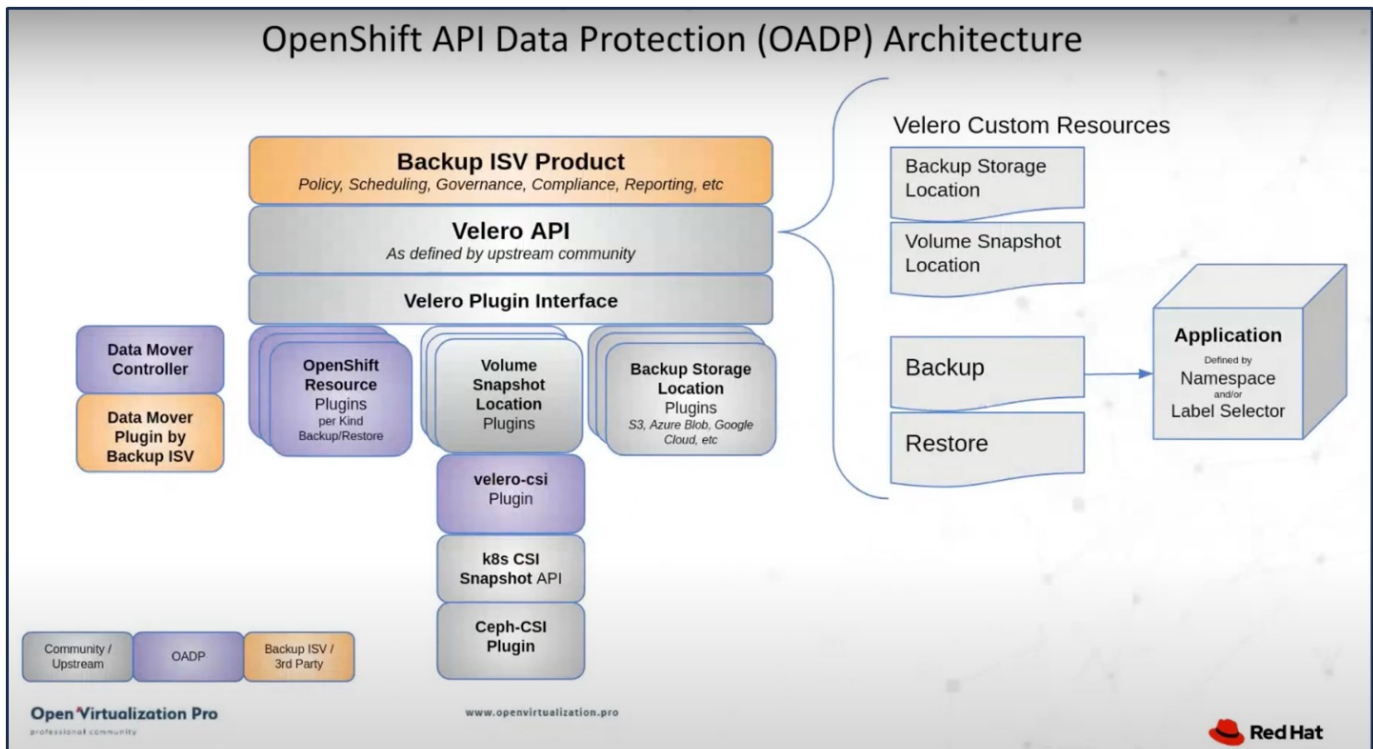
- ONTAP Object Storage
- StorageGrid

We then restore from the backup when needed. Please note that the app can be restored only to the cluster from where the backup was created.

OADP enables backup, restore, and disaster recovery of applications on an OpenShift cluster. Data that can be protected with OADP include Kubernetes resource objects, persistent volumes, and internal images.



Red Hat OpenShift has leveraged the solutions developed by the OpenSource communities for data protection. [Velero](#) is an open-source tool to safely backup and restore, perform disaster recovery, and migrate Kubernetes cluster resources and persistent volumes. To use Velero easily, OpenShift has developed the OADP operator and the Velero plugin to integrate with the CSI storage drivers. The core of the OADP APIs that are exposed are based on the Velero APIs. After installing the OADP operator and configuring it, the backup/restore operations that can be performed are based on the operations exposed by the Velero API.



OADP 1.3 is available from the operator hub of OpenShift cluster 4.12 and later. It has a built-in Data Mover that can move CSI volume snapshots to a remote object store. This provides portability and durability by moving snapshots to an object storage location during backup. The snapshots are then available for restoration after disasters.

The following are the versions of the various components used for the examples in this section

- OpenShift Cluster 4.14
- OADP Operator 1.13 provided by Red Hat
- Velero CLI 1.13 for Linux
- Astra Trident 24.02
- ONTAP 9.12
- postgresql installed using helm.

[Astra Trident CSI](#)  
[OpenShift API for Data Protection \(OADP\)](#)  
[Velero](#)

## Data protection for Container Apps in OpenShift Container Platform using OpenShift API for Data Protection (OADP)

Author: Banu Sundhar, NetApp

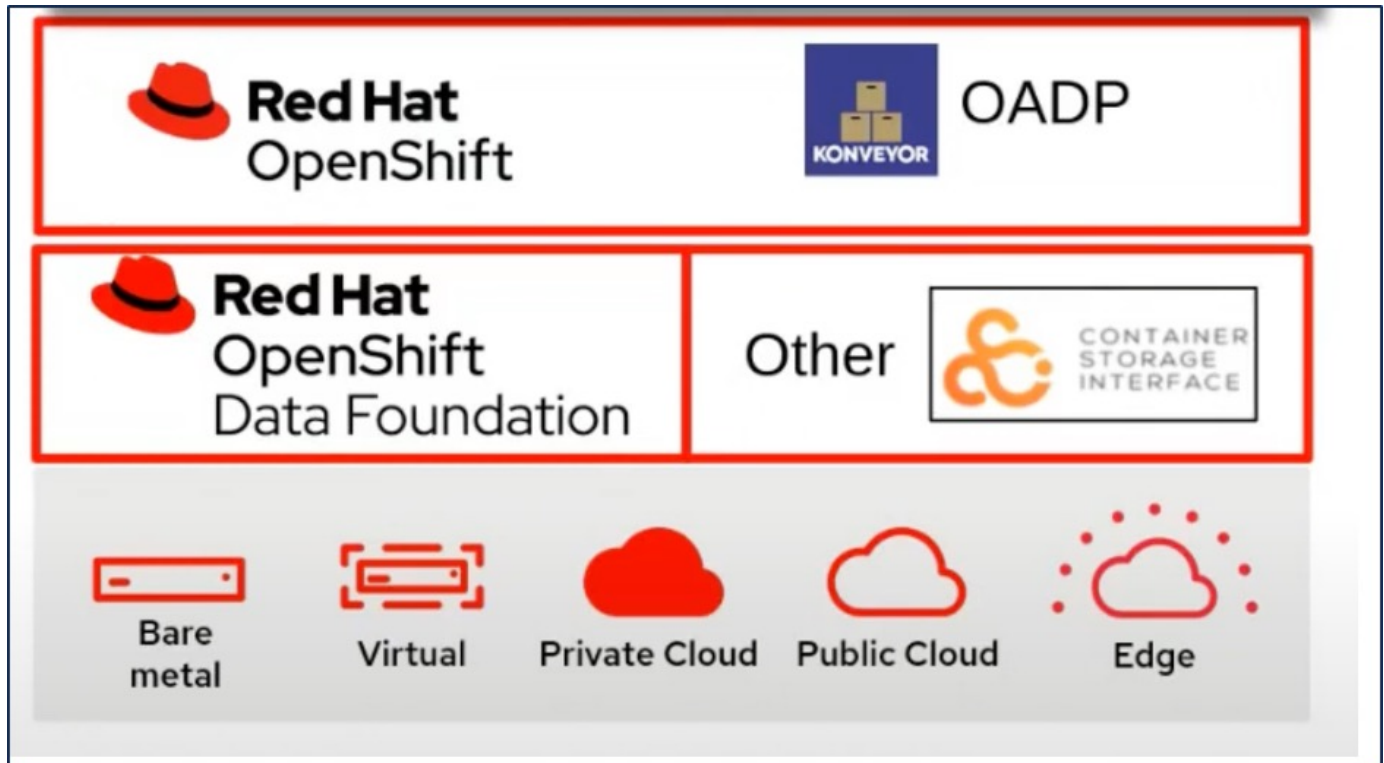
This section of the reference document provides details for creating backups of Container Apps using the OpenShift API for Data Protection (OADP) with Velero on NetApp ONTAP S3 or NetApp StorageGRID S3. The backups of namespace scoped resources including Persistent Volumes(PVs) of the app are created using CSI Astra Trident Snapshots.

The persistent storage for container apps can be backed by ONTAP storage integrated to the OpenShift Cluster using [Astra Trident CSI](#). In this section we use [OpenShift API for Data Protection \(OADP\)](#) to perform backup of apps including its data volumes to

- ONTAP Object Storage
- StorageGrid

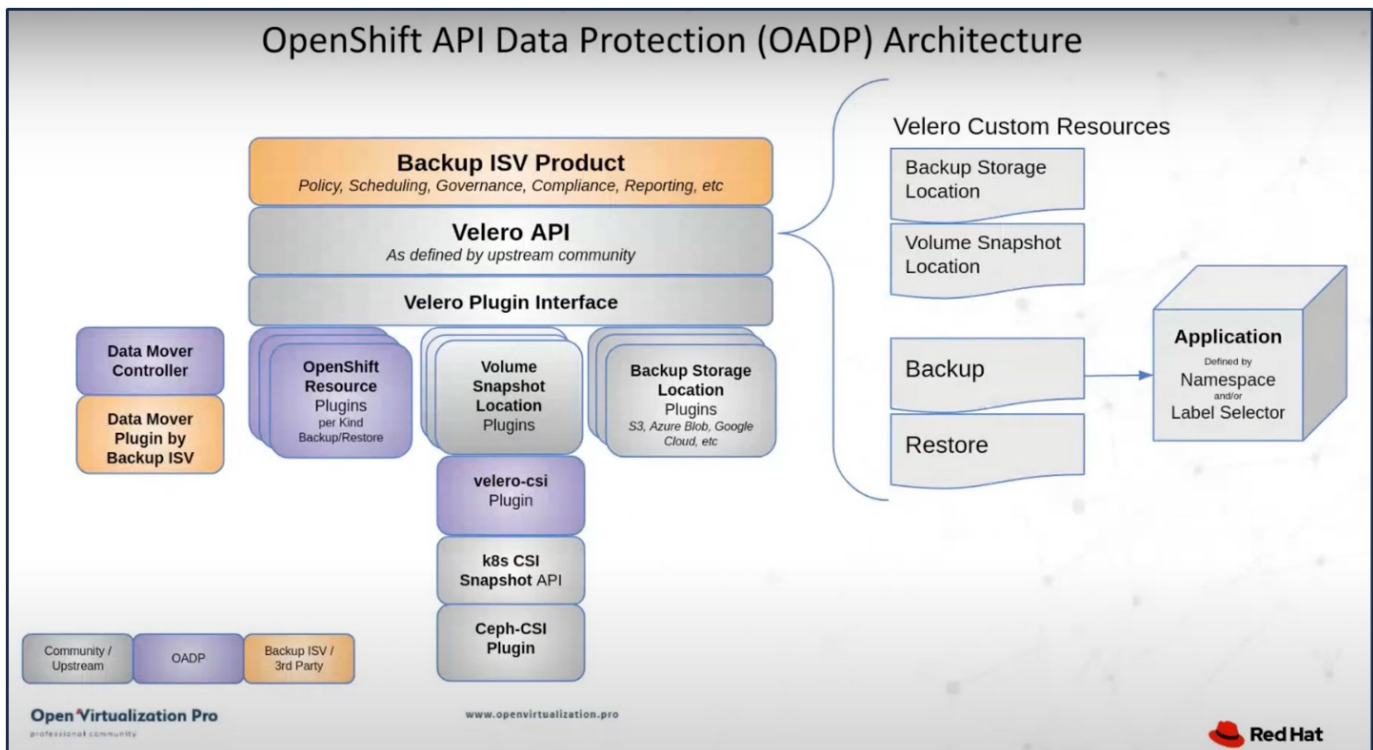
We then restore from the backup when needed. Please note that the app can be restored only to the cluster from where the backup was created.

OADP enables backup, restore, and disaster recovery of applications on an OpenShift cluster. Data that can be protected with OADP include Kubernetes resource objects, persistent volumes, and internal images.



Red Hat OpenShift has leveraged the solutions developed by the OpenSource communities for data protection. [Velero](#) is an open-source tool to safely backup and restore, perform disaster recovery, and migrate Kubernetes cluster resources and persistent volumes. To use Velero easily, OpenShift has developed the OADP operator and the Velero plugin to integrate with the CSI storage drivers. The core of the OADP APIs that are exposed are based on the Velero APIs. After installing the OADP operator and configuring it, the backup/restore operations that can be performed are based on the operations exposed by the Velero API.





OADP 1.3 is available from the operator hub of OpenShift cluster 4.12 and later. It has a built-in Data Mover that can move CSI volume snapshots to a remote object store. This provides portability and durability by moving snapshots to an object storage location during backup. The snapshots are then available for restoration after disasters.

The following are the versions of the various components used for the examples in this section

- OpenShift Cluster 4.14
- OADP Operator 1.13 provided by Red Hat
- Velero CLI 1.13 for Linux
- Astra Trident 24.02
- ONTAP 9.12
- postgresql installed using helm.

[Astra Trident CSI](#)  
[OpenShift API for Data Protection \(OADP\)](#)  
[Velero](#)

## Installation of OpenShift API for Data Protection (OADP) Operator

This section outlines the installation of OpenShift API for Data Protection (OADP) Operator.

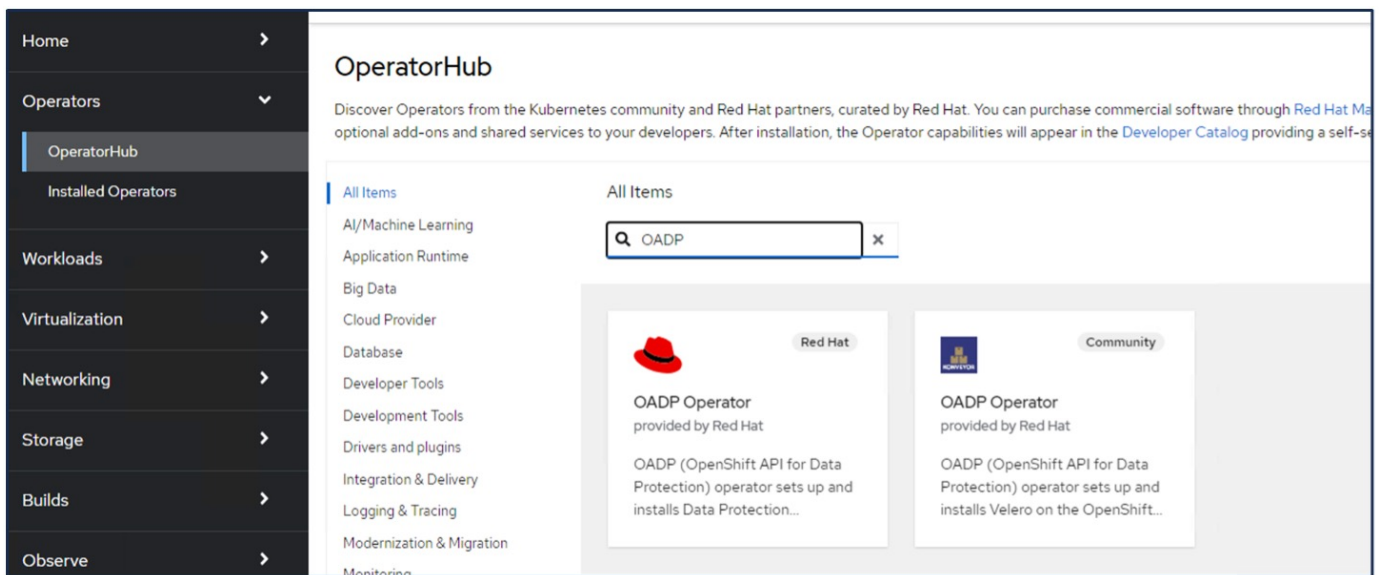
### Prerequisites

- A Red Hat OpenShift cluster (later than version 4.12) installed on bare-metal infrastructure with RHCOS worker nodes
- A NetApp ONTAP cluster integrated with the cluster using Astra Trident

- A Trident backend configured with an SVM on ONTAP cluster
- A StorageClass configured on the OpenShift cluster with Astra Trident as the provisioner
- Trident Snapshot class created on the cluster
- Cluster-admin access to Red Hat OpenShift cluster
- Admin access to NetApp ONTAP cluster
- An application eg. postgresql deployed on the cluster
- An admin workstation with tridentctl and oc tools installed and added to \$PATH

## Steps to install OADP Operator

1. Go to the Operator Hub of the cluster and select Red Hat OADP operator. In the Install page, use all the default selections and click install. On the next page, again use all the defaults and click Install. The OADP operator will be installed in the namespace openshift-adp.





# OADP Operator

1.3.0 provided by Red Hat

Install

## Channel

stable-1.3

## Version

1.3.0

## Capability level

- ☒ Basic Install
- ☒ Seamless Upgrades
- ☐ Full Lifecycle
- ☐ Deep Insights
- ☐ Auto Pilot

## Source

Red Hat

## Provider

Red Hat

## Infrastructure features

Disconnected

OpenShift API for Data Protection (OADP) operator sets up and installs Velero on the OpenShift platform, allowing users to backup and restore applications.

Backup and restore Kubernetes resources and internal images, at the granularity of a namespace, using a version of Velero appropriate for the installed version of OADP.

OADP backs up Kubernetes objects and internal images by saving them as an archive file on object storage. OADP backs up persistent volumes (PVs) by creating snapshots with the native cloud snapshot API or with the Container Storage Interface (CSI). For cloud providers that do not support snapshots, OADP backs up resources and PV data with Restic or Kopia.













- [Installing OADP for application backup and restore](#)
- [Installing OADP on a ROSA cluster and using STS, please follow the Getting Started Steps 1-3 in order to obtain the role ARN needed for using the standardized STS configuration flow via OLM](#)
- [Frequently Asked Questions](#)

Project: All Projects

## Installed Operators

Installed Operators are represented by ClusterServiceVersions within this Namespace. For more information, see the [Understanding Operators documentation](#) Operator and ClusterServiceVersion using the [Operator SDK](#).

Name Search by name...

Name	Namespace	Managed Namespaces	Status
 <b>OpenShift Virtualization</b> 4.14.4 provided by Red Hat	 openshift-cnv	 openshift-cnv	 Succeeded Up to date
 <b>OADP Operator</b> 1.3.0 provided by Red Hat	 openshift-adp	 openshift-adp	 Succeeded Up to date
 <b>Package Server</b> 0.0.1-snapshot provided by	 openshift-operator-lifecycle-manager	 openshift-operator-lifecycle-manager	 Succeeded



## Prerequisites for Velero configuration with Ontap S3 details

After the installation of the operator succeeds, configure the instance of Velero.

Velero can be configured to use S3 compatible Object Storage. Configure ONTAP S3 using the procedures shown in the [Object Storage Management section of ONTAP documentation](#). You will need the following information from your ONTAP S3 configuration to integrate with Velero.

- A Logical Interface (LIF) that can be used to access S3
- User credentials to access S3 that includes the access key and the secret access key
- A bucket name in S3 for backups with access permissions for the user
- For secure access to the Object storage, TLS certificate should be installed on the Object Storage server.

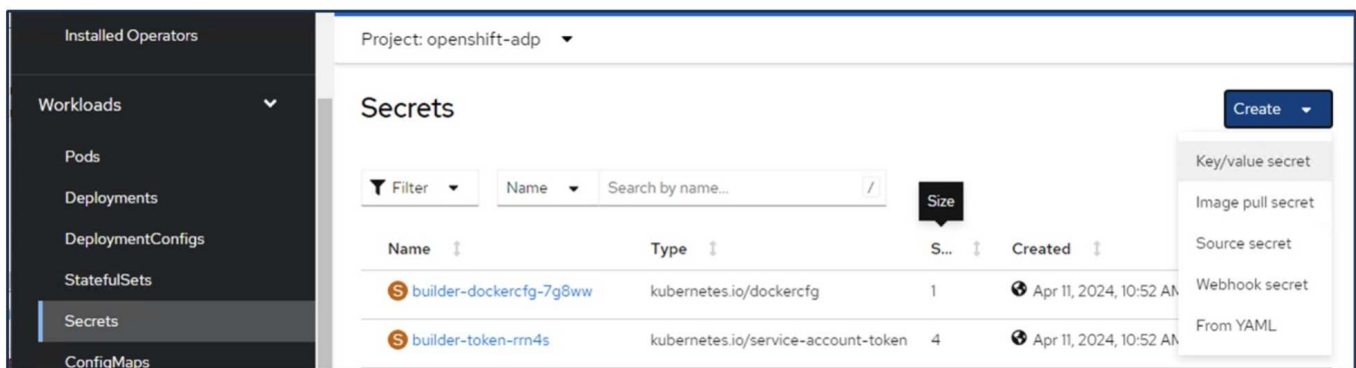
## Prerequisites for Velero configuration with StorageGrid S3 details

Velero can be configured to use S3 compatible Object Storage. You can configure StorageGrid S3 using the procedures shown in the [StorageGrid documentation](#). You will need the following information from your StorageGrid S3 configuration to integrate with Velero.

- The endpoint that can be used to access S3
- User credentials to access S3 that includes the access key and the secret access key
- A bucket name in S3 for backups with access permissions for the user
- For secure access to the Object storage, TLS certificate should be installed on the Object Storage server.

## Steps to configure Velero

- First, create a secret for an ONTAP S3 user credential or StorageGrid Tenant user credentials. This will be used to configure Velero later. You can create a secret from the CLI or from the web console. To create a secret from the web console, select Secrets, then click on Key/Value Secret. Provide the values for the credential name, key and the value as shown. Be sure to use the Access Key Id and Secret Access Key of your S3 user. Name the secret appropriately. In the sample below, a secret with ONTAP S3 user credentials named `ontap-s3-credentials` is created.



Project: openshift-adp ▼

---

## Edit key/value secret

Key/value secrets let you inject sensitive data into your application as files or environment variables.

**Secret name \***

ontap-s3-credentials

Unique name of the new secret.

**Key \***

cloud

**Value**

Browse...

Drag and drop file with your value here or browse to upload it.

```
[default]
aws_access_key_id=
aws_secret_access_key=
```

+ Add key/value

Save Cancel





To create a secret named sg-s3-credentials from the CLI you can use the following command.

```
# oc create secret generic sg-s3-credentials --namespace openshift-adp --from-file
cloud=cloud-credentials.txt
```

Where credentials.txt file contains the Access Key Id and the Secret Access Key of the S3 user in the following format:

```
[default]
aws_access_key_id=< Access Key ID of S3 user>
aws_secret_access_key=<Secret Access key of S3 user>|
```


- Next, to configure Velero, select Installed Operators from the menu item under Operators, click on OADP operator, and then select the **DataProtectionApplication** tab.

Home	Installed Operators				
Operators	Installed Operators are represented by ClusterServiceVersions within this Namespace. For more information, see the <a href="#">Understanding Operators documentation</a> or create an Operator and ClusterServiceVersion using the <a href="#">Operator SDK</a> .				
OperatorHub	<input type="text" value="Search by name..."/>				
Installed Operators					
Workloads					
Virtualization					
Networking					
	<b>Name</b>  <b>OADP Operator</b> 1.3.0 provided by Red Hat	<b>Managed Namespaces</b>  openshift-adp	<b>Status</b>  Succeeded Up to date	<b>Last updated</b>  Apr 11, 2024, 10:53 AM	<b>Provided APIs</b> <a href="#">BackupRepository</a> <a href="#">Backup</a> <a href="#">BackupStorageLocation</a> <a href="#">DeleteBackupRequest</a> <a href="#">View 11 more...</a>

Click on Create DataProtectionApplication. In the form view, provide a name for the DataProtection Application or use the default name.

Project: openshift-adp

Installed Operators > Operator details


**OADP Operator**  
1.3.0 provided by Red Hat

Actions

ServerStatusRequest
VolumeSnapshotLocation
DataDownload
DataUpload
CloudStorage
DataProtectionApplication

DataProtectionApplications

Create DataProtectionApplication

Now go to the YAML view and replace the spec information as shown in the yaml file examples below.

**Sample yaml file for configuring Velero with ONTAP S3 as the backupLocation**

```

spec:
  backupLocations:
    - velero:
        config:
          insecureSkipTLSVerify: 'false' ->use this for https
communication with ONTAP S3
          profile: default
          region: us-east-1
          s3ForcePathStyle: 'true' ->This allows use of IP in s3URL
          s3Url: 'https://10.61.181.161' ->Ensure TLS certificate for S3
is configured
          credential:
            key: cloud
            name: ontap-s3-credentials -> previously created secret
            default: true
          objectStorage:
            bucket: velero -> Your bucket name previously created in S3 for
backups
            prefix: container-demo-backup ->The folder that will be created
in the bucket
            caCert: <base64 encoded CA Certificate installed on ONTAP
Cluster with the SVM Scope where the bucker exists>
            provider: aws
          configuration:
            nodeAgent:
              enable: true
              uploaderType: kopia
              #default Data Mover uses Kopia to move snapshots to Object Storage
            velero:
              defaultPlugins:
                - csi ->This plugin to use CSI snapshots
                - openshift
                - aws
                - kubevirt -> This plugin to use Velero with OIpenShift
Virtualization

```

**Sample yaml file for configuring Velero with StorageGrid S3 as the backupLocation**

```
spec:
  backupLocations:
    - velero:
        config:
          insecureSkipTLSVerify: 'true'
          profile: default
          region: us-east-1 ->region of your StorageGrid system
          s3ForcePathStyle: 'True'
          s3Url: 'https://172.21.254.25:10443' ->the IP used to access S3
        credential:
          key: cloud
          name: sg-s3-credentials ->secret created earlier
        default: true
        objectStorage:
          bucket: velero
          prefix: demobackup
        provider: aws
  configuration:
    nodeAgent:
      enable: true
      uploaderType: kopia
    velero:
      defaultPlugins:
        - csi
        - openshift
        - aws
        - kubevirt
```

The spec section in the yaml file should be configured appropriately for the following parameters similar to the example above

### backupLocations

ONTAP S3 or StorageGrid S3 (with its credentials and other information as shown in the yaml) is configured as the default BackupLocation for velero.

### snapshotLocations

If you use Container Storage Interface (CSI) snapshots, you do not need to specify a snapshot location because you will create a VolumeSnapshotClass CR to register the CSI driver. In our example, you use Astra Trident CSI and you have previously created VolumeSnapShotClass CR using the Trident CSI driver.

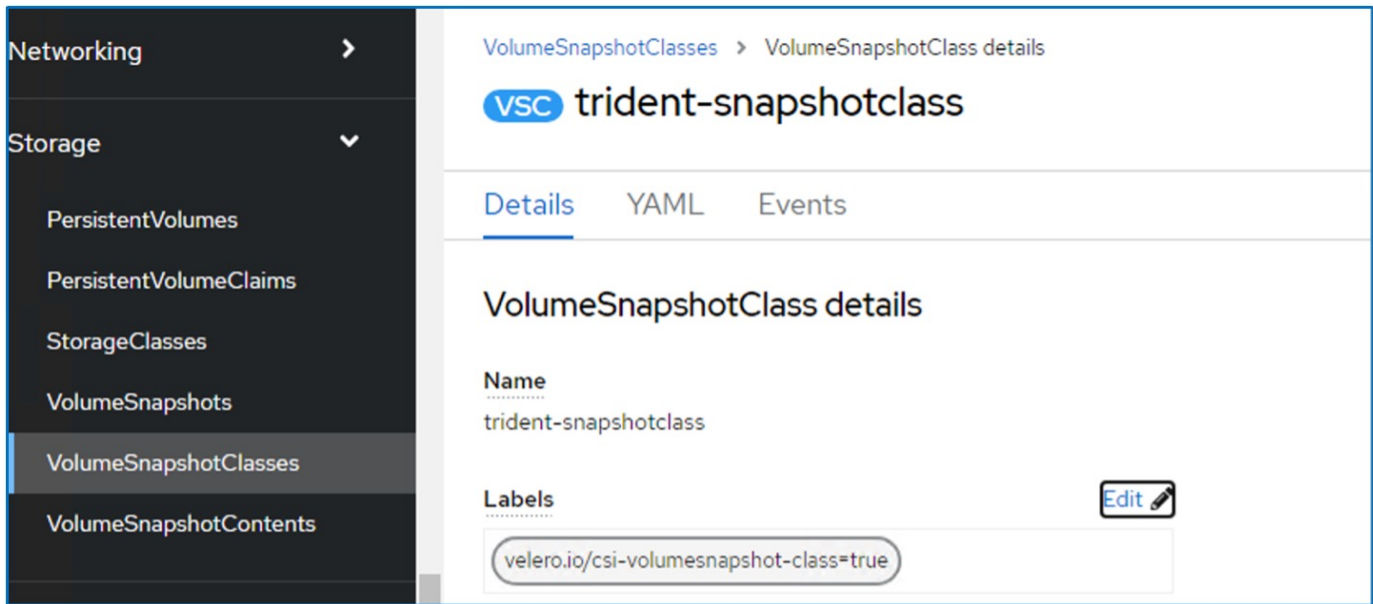
### Enable CSI plugin

Add csi to the defaultPlugins for Velero to back up persistent volumes with CSI snapshots.

The Velero CSI plugins, to backup CSI backed PVCs, will choose the VolumeSnapshotClass in the cluster that has **velero.io/csi-volumesnapshot-class** label set on it. For this

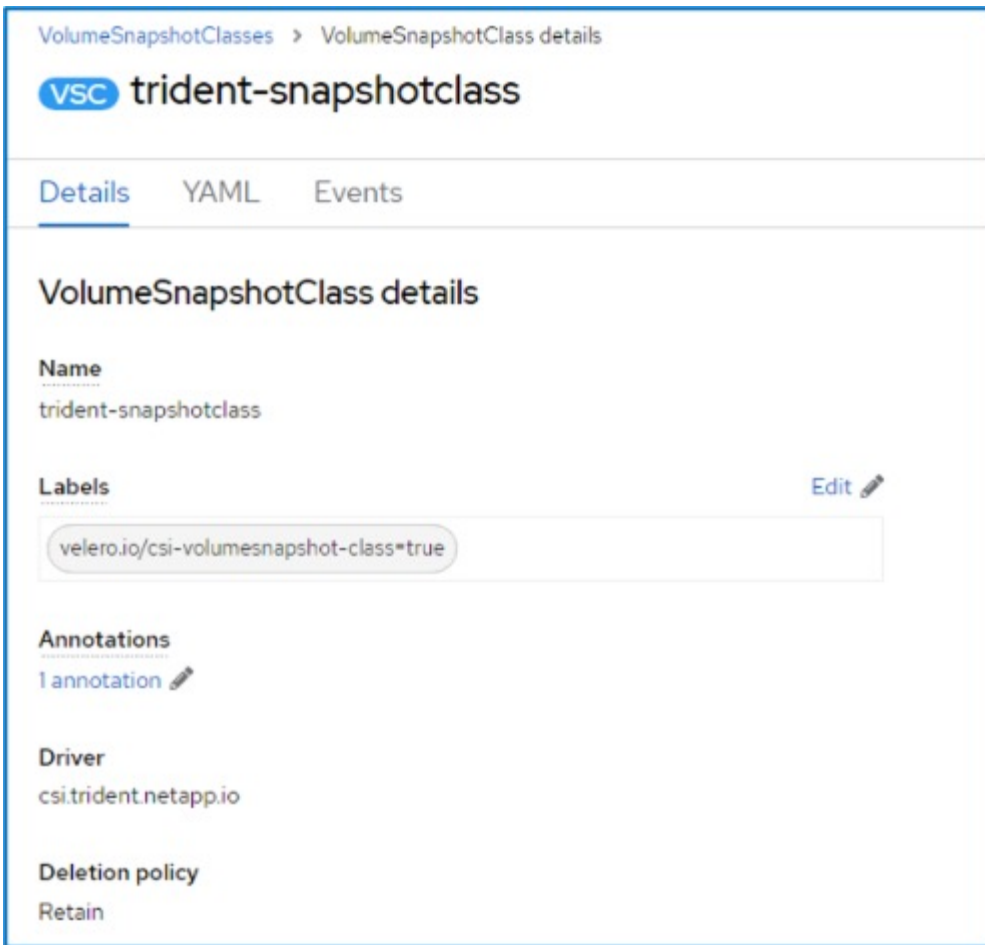
- You must have the trident VolumeSnapshotClass created.
- Edit the label of the trident-snapshotclass and set it to

`velero.io/csi-volumesnapshot-class=true` as shown below.

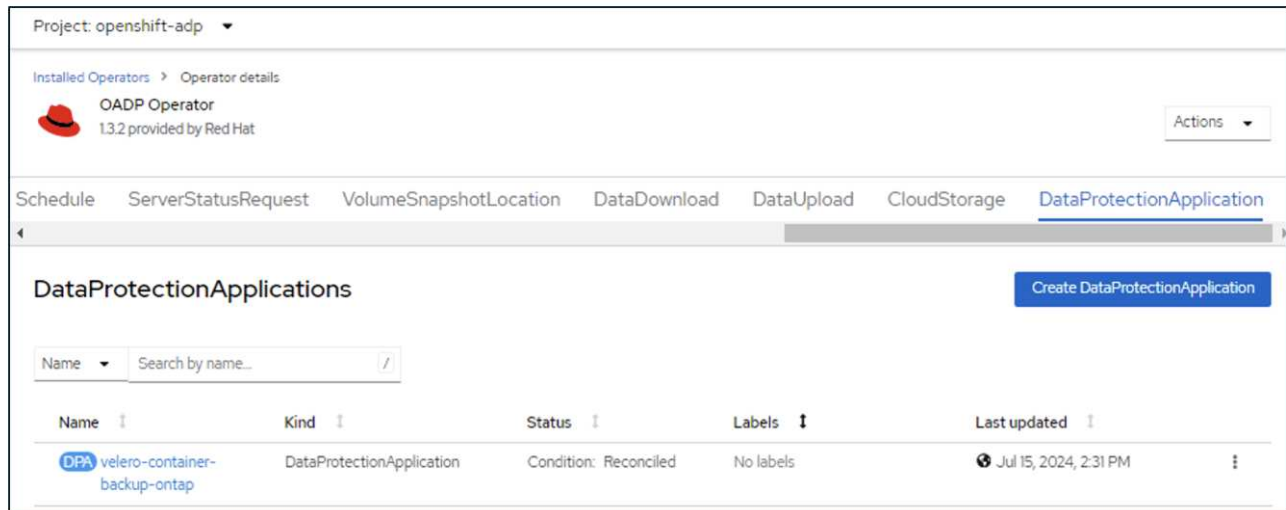


Ensure that the snapshots can persist even if the VolumeSnapshot objects are deleted. This can be done by setting the **deletionPolicy** to Retain. If not, deleting a namespace will completely lose all PVCs ever backed up in it.

```
apiVersion: snapshot.storage.k8s.io/v1
kind: VolumeSnapshotClass
metadata:
  name: trident-snapshotclass
driver: csi.trident.netapp.io
deletionPolicy: Retain
```




Ensure that the DataProtectionApplication is created and is in condition:Reconciled.



The OADP operator will create a corresponding BackupStorageLocation. This will be used when creating a backup.

Project: openshift-adp

Installed Operators > Operator details


**OADP Operator**  
1.3.2 provided by Red Hat

Actions

[BackupRepository](#)
[Backup](#)
[BackupStorageLocation](#)
[DeleteBackupRequest](#)
[DownloadRequest](#)
[PodVolumeBackup](#)
[PodVolumeRestore](#)

BSL

velero-container-backup-ontap-1

BackupStorageLocation

Phase: Available

app.kubernetes.io/component=bsl

app.kubernet...=velero-container...

app.kubernetes.io/m...=oadp-op...

app.kubernetes...=oadp-operato...

openshift.io/oadp=True

openshift.io/oadp-registry=True

Jul 15, 2024, 2:31 PM

Create BackupStorageLocation

Name

Search by name...

Name	Kind	Status	Labels	Last updated
BSL velero-container-backup-ontap-1	BackupStorageLocation	Phase: Available	<div>app.kubernetes.io/component=bsl</div> <div>app.kubernet...=velero-container...</div> <div>app.kubernetes.io/m...=oadp-op...</div> <div>app.kubernetes...=oadp-operato...</div> <div>openshift.io/oadp=True</div> <div>openshift.io/oadp-registry=True</div>	Jul 15, 2024, 2:31 PM

## Creating on-demand backup for Apps in OpenShift Container Platform

This section outlines how to create on-demand backup for VMs in OpenShift Virtualization.

### Steps to create a backup of an App

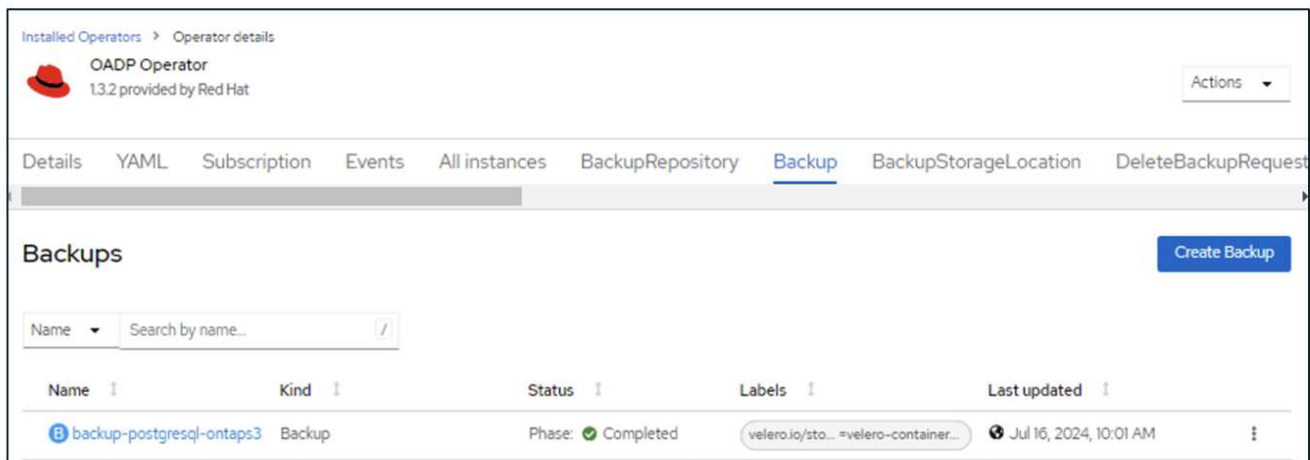
To create an on-demand backup of an app (app metadata and persistent volumes of the app), click on the **Backup** tab to create a Backup Custom Resource (CR). A sample yaml is provided to create the Backup CR. Using this yaml, the app and its persistent storage in the specified namespace will be backed up. Additional parameters can be set as shown in the [documentation](#).

A snapshot of the persistent volumes and the app resources in the namespace specified will be created by the CSI. This snapshot will be stored in the backup location specified in the yaml. The backup will remain in the system for 30 days as specified in the ttl.

```
spec:
  csiSnapshotTimeout: 10m0s
  defaultVolumesToFsBackup: false
  includedNamespaces:
    - postgresql ->namespace of the app
  itemOperationTimeout: 4h0m0s
  snapshotMoveData: false
  storageLocation: velero-container-backup-ontap-1 -->this is the
  backupStorageLocation previously created when Velero is configured.
  ttl: 720h0m0s
```

Once the backup completes, its Phase will show as completed.





You can inspect the backup in the Object storage with the help of an S3 browser application. The path of the backup shows up in the configured bucket with the prefix name (velero/container-demo-backup). You can see the contents of the backup includes the volume snapshots, logs, and other metadata of the application.



In StorageGrid, you can also use the S3 console that is available from the Tenant Manager to view the backup objects.

Path: / container-demo-backup/ backups/ backup-postgresql-ontaps3/

Name	Size	Type	Last Modified	Storage Class
backup-postgresql-ontaps3.tar.gz	384.66 KB	GZ File	7/16/2024 10:01:20 AM	STANDARD
velero-backup.json	3.30 KB	JSON File	7/16/2024 10:01:20 AM	STANDARD
backup-postgresql-ontaps3-csi-volumesnap...	731 bytes	GZ File	7/16/2024 10:01:19 AM	STANDARD
backup-postgresql-ontaps3-csi-volumesnap...	760 bytes	GZ File	7/16/2024 10:01:19 AM	STANDARD
backup-postgresql-ontaps3-resource-list.jso...	823 bytes	GZ File	7/16/2024 10:01:19 AM	STANDARD
backup-postgresql-ontaps3-itemoperations.j...	378 bytes	GZ File	7/16/2024 10:01:19 AM	STANDARD
backup-postgresql-ontaps3-volumesnapshot...	29 bytes	GZ File	7/16/2024 10:01:19 AM	STANDARD
backup-postgresql-ontaps3-podvolumeback...	29 bytes	GZ File	7/16/2024 10:01:19 AM	STANDARD
backup-postgresql-ontaps3-results.gz	49 bytes	GZ File	7/16/2024 10:01:19 AM	STANDARD
backup-postgresql-ontaps3-csi-volumesnap...	429 bytes	GZ File	7/16/2024 10:01:19 AM	STANDARD
backup-postgresql-ontaps3-logs.gz	12.01 KB	GZ File	7/16/2024 10:01:19 AM	STANDARD

Upload Download Delete New Folder Refresh

## Creating scheduled backups for Apps

To create backups on a schedule, you need to create a Schedule CR.

The schedule is simply a Cron expression allowing you to specify the time at which you want to create the backup. A sample yaml to create a Schedule CR is shown below.

```

apiVersion: velero.io/v1
kind: Schedule
metadata:
  name: schedule1
  namespace: openshift-adp
spec:
  schedule: 0 7 * * *
  template:
    includedNamespaces:
      - postgresql
    storageLocation: velero-container-backup-ontap-1

```

The Cron expression `0 7 * * *` means a backup will be created at 7:00 every day. The namespaces to be included in the backup and the storage location for the backup are also specified. So instead of a Backup CR, Schedule CR is used to create a backup at the specified time and frequency.

Once the schedule is created, it will be Enabled.


The screenshot shows the OADP Operator interface for the 'openshift-adp' project. The 'Schedule' tab is selected, displaying a table of schedules. A 'Create Schedule' button is visible in the top right. The table has columns for Name, Kind, Status, Labels, and Last updated. One schedule, 'schedule1', is listed with a status of 'Phase: Enabled' and a last updated time of 'Jul 16, 2024, 10:32 AM'.

Name	Kind	Status	Labels	Last updated
schedule1	Schedule	Phase:  Enabled	No labels	Jul 16, 2024, 10:32 AM

Backups will be created according to this schedule, and can be viewed from the Backup tab.

Project: openshift-adp

Installed Operators > Operator details


**OADP Operator**  
1.3.2 provided by Red Hat







Actions

All instances
BackupRepository
Backup
BackupStorageLocation
DeleteBackupRequest
DownloadRequest
PodVolumeBackup

Backups

Create Backup

Name
Search by name...

Name	Kind	Status	Labels	Last updated
 backup-postgresql-ontaps3	Backup	Phase:  Completed	velero.io/sto...=velero-container...	 Jul 16, 2024, 10:01 AM
 schedule1-20240717070005	Backup	Phase:  Completed	velero.io/schedule-na...=schedul... velero.io/sto...=velero-container...	 Jul 17, 2024, 3:00 AM

## Migrate an App from one cluster to another

Velero's backup and restore capabilities make it a valuable tool for migrating your data between clusters. This section describes how to migrate apps(s) from one cluster to another by creating a backup of the app in Object storage from one cluster and then restoring the app from the same object storage to another cluster. .

## Backup from first cluster

### Prerequisites on Cluster 1

- Astra Trident must be installed on the cluster.
- A trident backend and Storage class must be created.
- OADP operator must be installed on the cluster.
- The DataProtectionApplication should be configured.

Use the following spec to configure the DataProtectionApplication object.

```
spec:
  backupLocations:
    - velero:
        config:
          insecureSkipTLSVerify: 'false'
          profile: default
          region: us-east-1
          s3ForcePathStyle: 'true'
          s3Url: 'https://10.61.181.161'
        credential:
          key: cloud
          name: ontap-s3-credentials
        default: true
        objectStorage:
          bucket: velero
          caCert: <base-64 encoded tls certificate>
          prefix: container-backup
        provider: aws
  configuration:
    nodeAgent:
      enable: true
      uploaderType: kopia
    velero:
      defaultPlugins:
        - csi
        - openshift
        - aws
        - kubevirt
```

- Create an application on the cluster and take a backup of this application. As an example, install a postgres application.


```
[root@localhost ~]# oc get nodes
NAME                STATUS    ROLES    AGE      VERSION
ocp6-master1        Ready     control-plane,master 3d13h    v1.27.15+6147456
ocp6-master2        Ready     worker    3d12h    v1.27.15+6147456
ocp6-master3        Ready     control-plane,master 3d13h    v1.27.15+6147456
ocp6-worker1        Ready     worker    3d12h    v1.27.15+6147456
ocp6-worker2        Ready     worker    3d12h    v1.27.15+6147456
ocp6-worker3        Ready     control-plane,master 3d12h    v1.27.15+6147456
[root@localhost ~]# helm install postgresql bitnami/postgresql -n postgresql --create namespace^C
[root@localhost ~]# oc get pods -n postgresql
NAME                READY    STATUS    RESTARTS   AGE
postgresql-0        1/1      Running   0           4h53m
[root@localhost ~]# oc get pvc -n postgresql
NAME                STATUS    VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS   AGE
data-postgresql-0   Bound     pvc-f7a3c772-0e61-49cb-a3d0-7c7b2ec87dc6 8Gi        RWO            ontap-nas      4h53m
[root@localhost ~]# oc get pv -n postgresql
NAME                CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS   CLAIM                                STORAGECLASS
REASON    AGE
pvc-2e9e982f-54a4-4e7b-8eae-a589e0d9d819 1Gi       RWO            Delete            Bound    trident/basic                        ontap-nas
4h55m
pvc-f7a3c772-0e61-49cb-a3d0-7c7b2ec87dc6 8Gi       RWO            Delete            Bound    postgresql/data-postgresql-0        ontap-nas
4h53m
[root@localhost ~]#
```

- Use the following spec for the backup CR:

```
spec:
  csiSnapshotTimeout: 10m0s
  defaultVolumesToFsBackup: false
  includedNamespaces:
    - postgresql
  itemOperationTimeout: 4h0m0s
  snapshotMoveData: true
  storageLocation: velero-sample-1
  ttl: 720h0m0s
```

Project: openshift-adp ▾

Installed Operators > Operator details



 **OADP Operator**  
1.4.0 provided by Red Hat

Actions ▾

Repository Backup BackupStorageLocation DeleteBackupRequest DownloadRequest PodVolumeBackup PodVolumeRes

**Backups** Create Backup

Name ▾ Search by name... /

Name	Kind	Status
 backup	Backup	Phase:  Completed

Go to Settings to activate Windows.

You can click on the **All instances** tab to see the different objects being created and moving through different phases to finally come to the backup **completed** phase.

A backup of the resources in the namespace postgresql will be stored in the Object Storage location (ONTAP S3) specified in the backupLocation in the OADP spec.

## Restore to a second cluster

### Prerequisites on Cluster 2


- Astra Trident must be installed on cluster 2.
- The postgresql app must NOT be already installed in the postgresql namespace.
- OADP operator must be installed on cluster 2, and the BackupStorage Location must be pointing to the same object storage location where the backup was stored from the first cluster.
- The Backup CR must be visible from the second cluster.

```
[root@localhost ~]# oc get pods -n trident
NAME                                READY   STATUS    RESTARTS   AGE
trident-controller-6799cfb77f-8rzvk 6/6     Running   6           2d7h
trident-node-linux-7wvjz            2/2     Running   2           2d7h
trident-node-linux-8vvm2            2/2     Running   0           2d7h
trident-node-linux-bgs6f            2/2     Running   2           2d7h
trident-node-linux-njwb8            2/2     Running   0           2d7h
trident-node-linux-scqjl            2/2     Running   0           2d7h
trident-node-linux-swr69            2/2     Running   2           2d7h
trident-operator-b88b86fc8-7fk68    1/1     Running   1           2d7h
[root@localhost ~]#
```

```
[root@localhost ~]# oc get nodes
NAME                STATUS    ROLES    AGE   VERSION
ocp7-master1        Ready     control-plane,master 3d    v1.27.15+6147456
ocp7-master2        Ready     control-plane,master 3d    v1.27.15+6147456
ocp7-master3        Ready     control-plane,master 3d    v1.27.15+6147456
ocp7-worker1        Ready     worker    3d    v1.27.15+6147456
ocp7-worker2        Ready     worker    3d    v1.27.15+6147456
ocp7-worker3        Ready     worker    3d    v1.27.15+6147456
[root@localhost ~]# oc get pods -n postgresql
No resources found in postgresql namespace.
[root@localhost ~]# oc get pvc -n postgresql
No resources found in postgresql namespace.
[root@localhost ~]# oc get pv -n postgresql
NAME                CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS   CLAIM                STORAGECLASS   REASON   AGE
pvc-c6660630-0cfe-484b-aaa3-5ada54c8b9a7 1Gi        RWO            Delete           Bound    trident/basic        OnTarMag 11m
pvc-edcc6551-81b0-40b4-8547-e9df70c1740d 10Gi       RWO            Delete           Bound    default/test-pvc     vsphere-sc   2d7h
n
[root@localhost ~]#
```

Project: openshift-adp

Installed Operators > Operator details

 OADP Operator

1.4.0 provided by Red Hat


Actions

BackupBackupStorageLocationDeleteBackupRequestDownloadRequestPodVolumeBackupPodVolumeRestoreRes


BackupStorageLocations

Create BackupStorageLocation

NameSearch by name.../

Name	Kind	Status
 velero-container-demo-1	BackupStorageLocation	Phase: Available

Installed Operators > Operator details



OADP Operator

1.4.0 provided by Red Hat

Actions

Details

YAML

Subscription

Events

All instances

BackupRepository

Backup

BackupStorageLocation

DeleteBackupRequest




DownloadRequest

Backups

Create Backup

Name

Search by name...

Name	Kind	Status	Labels	Last updated
 backup	Backup	Phase:  Completed	velero.io/storage-locati...=velero-sampl...	 Jul 25, 2024, 8:39 PM

Restore the app on this cluster from the backup. Use the following yaml to create the Restore CR.

```


apiVersion: velero.io/v1
kind: Restore
apiVersion: velero.io/v1
metadata:
  name: restore
  namespace: openshift-adp
spec:
  backupName: backup
  restorePVs: true

```

When the restore is completed, you will see that the postgresql app is running on this cluster and is associated with the pvc and a corresponding pv. The state of the app is the same as when the backup was taken.

Project: openshift-adp

Installed Operators > Operator details



OADP Operator

1.4.0 provided by Red Hat

Actions

eLocation

DeleteBackupRequest

DownloadRequest

PodVolumeBackup

PodVolumeRestore

Restore

Schedule



Server

Restores

Create Restore

Name

Search by name...

Name	Kind	Status
 restore	Restore	Phase:  Completed



```
[root@localhost ~]# export KUBECONFIG=ocp-cluster7/kubeconfig-ocp-cluster7
[root@localhost ~]# oc get nodes
NAME                STATUS    ROLES                  AGE      VERSION
ocp7-master1        Ready     control-plane,master   3d3h     v1.27.15+6147456
ocp7-master2        Ready     control-plane,master   3d3h     v1.27.15+6147456
ocp7-master3        Ready     control-plane,master   3d3h     v1.27.15+6147456
ocp7-worker1        Ready     worker                 3d3h     v1.27.15+6147456
ocp7-worker2        Ready     worker                 3d3h     v1.27.15+6147456
ocp7-worker3        Ready     worker                 3d3h     v1.27.15+6147456
[root@localhost ~]# oc get pods -n postgresql
NAME                READY    STATUS    RESTARTS   AGE
postgresql-0        1/1      Running   0           31m
[root@localhost ~]# oc get pvc -n postgresql
NAME                STATUS    VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS   AGE
data-postgresql-0    Bound     pvc-ce7044e3-2ba5-4934-8bad-553fa7d35128  8Gi        RWO            ontap-nas      31m
[root@localhost ~]# oc get pv
NAME                CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS   CLAIM                STORAGECLASS
REASON             AGE
pvc-c6660630-0cfe-484b-aaa3-5ada54c8b9a7  1Gi        RWO            Delete           Bound    trident/basic        ontap-nas
3h27m
pvc-ce7044e3-2ba5-4934-8bad-553fa7d35128  8Gi        RWO            Delete           Bound    postgresql/data-postgresql-0  ontap-nas
31m
pvc-edcc6551-81b0-40b4-8547-e9df70c1740d  10Gi       RWO            Delete           Bound    default/test-pvc       vsphere-sc
2d10h
[root@localhost ~]#
```

## Restore an App from a backup

This section describes how to restore apps(s) from a backup.

### Prerequisites

To restore from a backup, let us assume that the namespace where the app existed got accidentally deleted.

```
[root@localhost ~]# oc get pods -n postgresql
NAME                READY    STATUS    RESTARTS   AGE
postgresql-0        1/1      Running   0           102s
[root@localhost ~]# oc delete ns postgresql
namespace "postgresql" deleted


[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# oc get pods -n postgresql
No resources found in postgresql namespace.
[root@localhost ~]#
```

## Restore to the same namespace

To restore from the backup that we just created, we need to create a Restore Custom Resource (CR). We need to provide it a name, provide the name of the backup that we want to restore from and set the restorePVs to true. Additional parameters can be set as shown in the [documentation](#). Click on Create button.

Project: openshift-adp ▾

Installed Operators > Operator details

 **OADP Operator**  
1.3.0 provided by Red Hat

Actions ▾

est

DownloadRequest

PodVolumeBackup

PodVolumeRestore

**Restore**

Schedule

ServerStatusRequest

VolumeSnap

Restores


Create Restore

```
apiVersion: velero.io/v1
kind: Restore
apiVersion: velero.io/v1
metadata:
  name: restore
  namespace: openshift-adp
spec:
  backupName: backup-postgresql-ontaps3
  restorePVs: true
```

When the phase shows completed, you can see that the app has been restored to the state when the snapshot was taken. The app is restored to the same namespace.

Project: openshift-adp ▾

Installed Operators > Operator details

 **OADP Operator**  
1.3.0 provided by Red Hat

Actions ▾

est

DownloadRequest

PodVolumeBackup

PodVolumeRestore

**Restore**

Schedule

ServerStatusRequest



VolumeSr

Restores

Create Restore

Name ▾

Search by name... /

Name ▴	Kind ▴	Status ▴	Labels ▴
 restore1	Restore	Phase:  Completed	No labels

```

[root@localhost ~]#
[root@localhost ~]# oc get pods -n postgresql
No resources found in postgresql namespace.
[root@localhost ~]# oc get pods -n postgresql
NAME          READY   STATUS             RESTARTS   AGE
postgresql-0   0/1     ContainerCreating   0           16s
[root@localhost ~]# oc get pods -n postgresql
NAME          READY   STATUS    RESTARTS   AGE
postgresql-0   0/1     Running   0           22s
[root@localhost ~]# oc get pods -n postgresql
NAME          READY   STATUS    RESTARTS   AGE
postgresql-0   0/1     Running   0           29s
[root@localhost ~]# oc get pods -n postgresql
NAME          READY   STATUS    RESTARTS   AGE
postgresql-0   1/1     Running   0           37s
[root@localhost ~]#

```

## Restore to a different namespace

To restore the App to a different namespace, you can provide a namespaceMapping in the yaml definition of the Restore CR.

The following sample yaml file creates a Restore CR to restore an App and its persistent storage from the postgresql namespace, to the new namespace postgresql-restored.

```
apiVersion: velero.io/v1
kind: Restore
metadata:
  name: restore-to-different-ns
  namespace: openshift-adp
spec:
  backupName: backup-postgresql-ontaps3
  restorePVs: true
  includedNamespaces:
  - postgresql
  namespaceMapping:
    postgresql: postgresql-restored
```

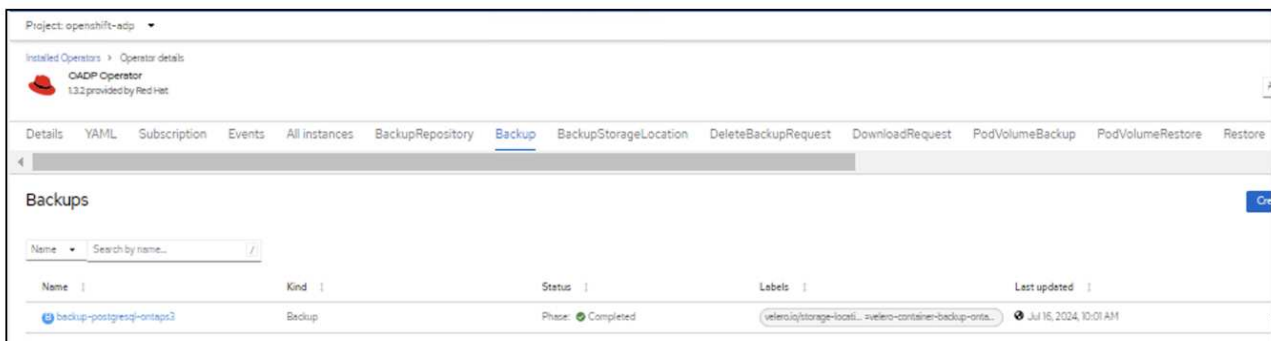
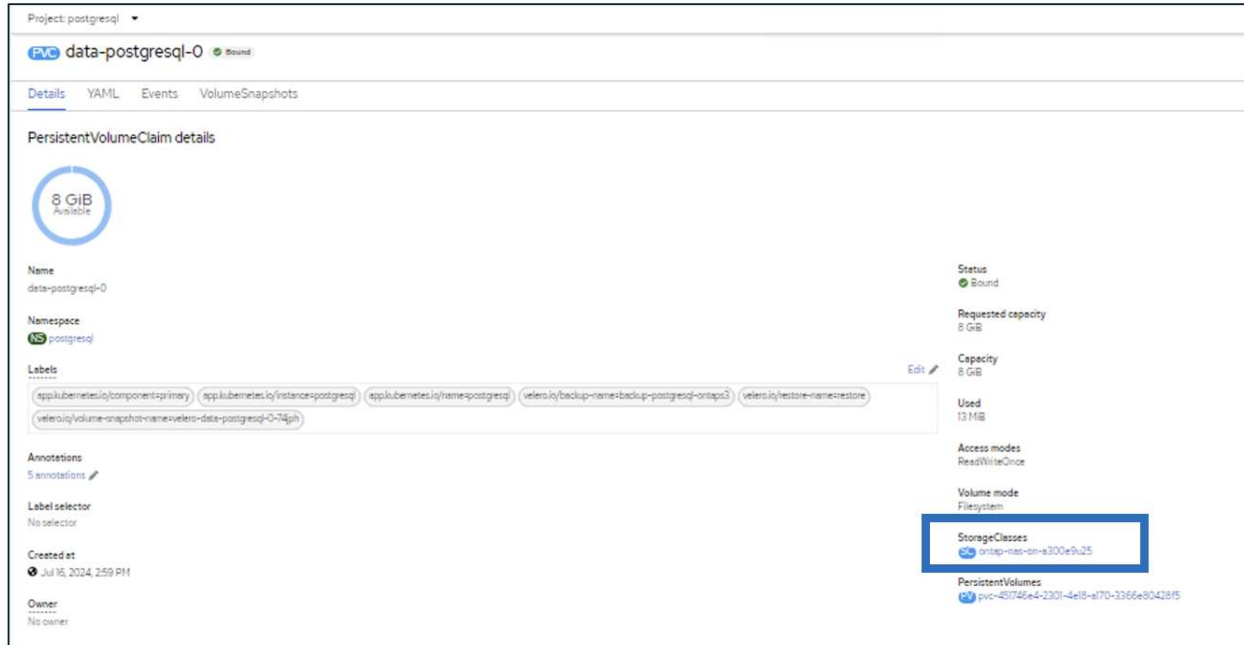
When the phase shows completed, you can see that the app has been restored to the state when the snapshot was taken. The App is restored to a different namespace as specified in the yaml.

```
[root@localhost ~]# oc get pods -n postgresql
No resources found in postgresql namespace.
[root@localhost ~]# oc get pods -n postgresql-restored
NAME                READY   STATUS    RESTARTS   AGE
postgresql-0        0/1     Running   0           19s
[root@localhost ~]# oc get pods -n postgresql-restored
NAME                READY   STATUS    RESTARTS   AGE
postgresql-0        0/1     Running   0           22s
[root@localhost ~]# oc get pods -n postgresql-restored
NAME                READY   STATUS    RESTARTS   AGE
postgresql-0        1/1     Running   0           36s
[root@localhost ~]#
```

## Restore to a different storage class

Velero provides a generic ability to modify the resources during restore by specifying json patches. The json patches are applied to the resources before they are restored. The json patches are specified in a configmap and the configmap is referenced in the restore command. This feature enables you to restore using different storage class.

In the example below, the app, during deployment uses ontap-nas as the storage class for its persistent volumes. A backup of the app named backup-postgresql-ontaps3 is created.



Simulate a loss of the app by uninstalling the app.

To restore the VM using a different storage class, for example, ontap-nas-eco storage class, you need to do the following two steps:

### Step 1

Create a config map (console) in the openshift-adp namespace as follows:

Fill in the details as shown in the screenshot:

Select namespace : openshift-adp

Name: change-ontap-sc (can be any name)

Key: change-ontap-sc-config.yaml:

Value:

```
version: v1
resourceModifierRules:
- conditions:
    groupResource: persistentvolumeclaims
    resourceNameRegex: "data-postgresql*"
    namespaces:
    - postgresql
patches:
- operation: replace
  path: "/spec/storageClassName"
  value: "ontap-nas-eco"
```

Project: openshift-adp ▾

## Edit ConfigMap

Config maps hold key-value pairs that can be used in pods to read application configuration.

Configure via: ☒ Form view ☐ YAML view

**Name \***

change-ontap-sc

A unique name for the ConfigMap within the project

☐ Immutable

Immutable, if set to true, ensures that data stored in the ConfigMap cannot be updated

**Data**

Data contains the configuration data that is in UTF-8 range

[Remove key/value](#)

**Key \***

change-ontap-sc.yaml

**Value**

Browse...

Drag and drop file with your value here or browse to upload it.

```
version: v1
resourceModifierRules:
- conditions:
    groupResource: persistentvolumeclaims
    resourceNameRegex: "data-postgresql*"
    namespaces:
    - postgresql
  patches:
  - operation: replace
    path: "/spec/storageClassName"
    value: "ontap-nas-eco"
```

The resulting config map object should look like this (CLI):



```
[root@localhost ~]# kubectl describe cm/change-ontap-sc -n openshift-adp
Name:          change-ontap-sc
Namespace:     openshift-adp
Labels:        <none>
Annotations:    <none>

Data
====
change-ontap-sc.yaml:
----
version: v1
resourceModifierRules:
- conditions:
    groupResource: persistentvolumeclaims
    resourceNameRegex: "data-postgresql*"
    namespaces:
    - postgresql
  patches:
  - operation: replace
    path: "/spec/storageClassName"
    value: "ontap-nas-eco"

BinaryData
====

Events: <none>
[root@localhost ~]#
```

This config map will apply the resource modifier rule when the restore is created. A patch will be applied to replace the storage class name to ontap-nas-eco for all persistent volume claims starting with rhel.

## Step 2

To restore the VM use the following command from the Velero CLI:

```
#velero restore create restore1 --from-backup backup1 --resource
-modifier-configmap change-storage-class-config -n openshift-adp
```

The app is restored in the same namespace with the persistent volume claims created using the storage class ontap-nas-eco.

```
[root@localhost ~]# oc get pods -n postgresql
NAME          READY   STATUS    RESTARTS   AGE
postgresql-0  1/1     Running   0           11m
[root@localhost ~]# oc get pvc -n postgresql
NAME          STATUS    VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS   AGE
data-postgresql-0  Bound    pvc-33526ea4-37c2-4180-a9f6-fb47aea3b4e2  8Gi        RWO            ontap-nas-eco  11m
[root@localhost ~]#
```

## Deleting backups and restores in using Velero

This section outlines how to delete backups and restores of Apps in OpenShift container platform using Velero.

### List all backups

You can list all Backup CRs by using the OC CLI tool or the Velero CLI tool.

Download the Velero CLI as given in the instructions in the [Velero documentation](#).

```
[root@localhost ~]# oc get backups -n openshift-adp
NAME                                AGE
backup-postgresql-ontaps3          23h
backup2                             26s
schedule1-20240717070005           6h42m
[root@localhost ~]# velero get backups -n openshift-adp
NAME                                STATUS  ERRORS  WARNINGS  CREATED                                EXPIRES  STORAGE LOCATION  SELECTOR
backup-postgresql-ontaps3          Completed  0        0         2024-07-16 10:01:08 -0400 EDT        29d      velero-container-backup-ontap-1  <none>
backup2                             Completed  0        0         2024-07-17 09:42:32 -0400 EDT        29d      velero-container-backup-ontap-1  <none>
schedule1-20240717070005          Completed  0        0         2024-07-17 03:00:05 -0400 EDT        29d      velero-container-backup-ontap-1  <none>
[root@localhost ~]#
```

### Deleting a backup

You can delete a Backup CR without deleting the Object Storage data by using the OC CLI tool. The backup will be removed from the CLI/Console output. However, since the corresponding backup is not removed from the object storage, it will re-appear in the CLI/console output.

```
[root@localhost ~]# oc delete backup backup2 -n openshift-adp
backup.velero.io "backup2" deleted
[root@localhost ~]# oc get backups -n openshift-adp
NAME                                AGE
backup-postgresql-ontaps3          23h
schedule1-20240717070005           6h49m
[root@localhost ~]# oc get backups -n openshift-adp
NAME                                AGE
backup-postgresql-ontaps3          23h
backup2                             24s
schedule1-20240717070005           6h50m
[root@localhost ~]#
```

If you want to delete the Backup CR AND the associated object storage data, you can do so by using the Velero CLI tool.

```
[root@localhost ~]# velero get backups -n openshift-adp
```

NAME	STATUS	ERRORS	WARNINGS	CREATED	EXPIRES	STORAGE LOCATION	SELECTOR
backup-postgresql-ontaps3	Completed	0	0	2024-07-16 10:01:08 -0400 EDT	29d	velero-container-backup-ontap-1	<none>
backup2	Completed	0	0	2024-07-17 09:42:32 -0400 EDT	29d	velero-container-backup-ontap-1	<none>
schedule1-20240717070005	Completed	0	0	2024-07-17 03:00:05 -0400 EDT	29d	velero-container-backup-ontap-1	<none>

```
[root@localhost ~]# velero delete backup backup2 -n openshift-adp
Are you sure you want to continue (Y/N)? Y
Request to delete backup "backup2" submitted successfully.
The backup will be fully deleted after all associated data (disk snapshots, backup files, restores) are removed.
[root@localhost ~]# velero get backups -n openshift-adp
```

NAME	STATUS	ERRORS	WARNINGS	CREATED	EXPIRES	STORAGE LOCATION	SELECTOR
backup-postgresql-ontaps3	Completed	0	0	2024-07-16 10:01:08 -0400 EDT	29d	velero-container-backup-ontap-1	<none>
schedule1-20240717070005	Completed	0	0	2024-07-17 03:00:05 -0400 EDT	29d	velero-container-backup-ontap-1	<none>

```
[root@localhost ~]#
```

## Deleting the Restore

You can delete the Restore CR Object by using either the OC CLI or the Velero CLI

```
[root@localhost ~]# velero get restore -n openshift-adp
```

NAME	BACKUP	STATUS	STARTED	COMPLETED	ERRORS	WARNINGS	CREATED	SELECTOR
restore	backup-postgresql-ontaps3	Completed	2024-07-16 14:59:22 -0400 EDT	2024-07-16 14:59:45 -0400 EDT	0	10	2024-07-16 14:59:22 -0400 EDT	<none>
restore1	backup-postgresql-ontaps3	Completed	2024-07-16 16:36:37 -0400 EDT	2024-07-16 16:36:59 -0400 EDT	0	9	2024-07-16 16:36:37 -0400 EDT	<none>

```
[root@localhost ~]# velero restore delete restore1 -n openshift-adp
Are you sure you want to continue (Y/N)? Y
Request to delete restore "restore1" submitted successfully.
The restore will be fully deleted after all associated data (restore files in object storage) are removed.
[root@localhost ~]# velero get restore -n openshift-adp
```

NAME	BACKUP	STATUS	STARTED	COMPLETED	ERRORS	WARNINGS	CREATED	SELECTOR
restore	backup-postgresql-ontaps3	Completed	2024-07-16 14:59:22 -0400 EDT	2024-07-16 14:59:45 -0400 EDT	0	10	2024-07-16 14:59:22 -0400 EDT	<none>

```
[root@localhost ~]#
[root@localhost ~]# oc delete restore restore -n openshift-adp
restore.velero.io "restore" deleted
[root@localhost ~]# oc get restore -n openshift-adp
No resources found in openshift-adp namespace.
[root@localhost ~]# velero get restore -n openshift-adp
[root@localhost ~]#
```

Activate Windows

## Copyright information

Copyright © 2024 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.