



# **Red Hat OpenShift Virtualization with NetApp ONTAP**

NetApp Solutions

NetApp  
April 26, 2024

This PDF was generated from [https://docs.netapp.com/us-en/netapp-solutions/containers/rh-os-n\\_use\\_case\\_openshift\\_virtualization\\_overview.html](https://docs.netapp.com/us-en/netapp-solutions/containers/rh-os-n_use_case_openshift_virtualization_overview.html) on April 26, 2024. Always check docs.netapp.com for the latest.

# Table of Contents

- Red Hat OpenShift Virtualization with NetApp ONTAP ..... 1
  - Red Hat OpenShift Virtualization with NetApp ONTAP ..... 1
  - Deployment for OpenShift Virtualization ..... 1
  - Data Protection for OpenShift Virtualization..... 26
  - Monitoring using Cloud Insights ..... 48

# Red Hat OpenShift Virtualization with NetApp ONTAP

## Red Hat OpenShift Virtualization with NetApp ONTAP

Depending on the specific use case, both containers and virtual machines (VMs) can serve as optimal platforms for different types of applications. Therefore, many organizations run some of their workloads on containers and some on VMs. Often, this leads organizations to face additional challenges by having to manage separate platforms: a hypervisor for VMs and a container orchestrator for applications.

To address this challenge, Red Hat introduced OpenShift Virtualization (formerly known as Container Native Virtualization) starting from OpenShift version 4.6. The OpenShift Virtualization feature enables you to run and manage virtual machines alongside containers on the same OpenShift Container Platform installation, providing hybrid management capability to automate deployment and management of VMs through operators. In addition to creating VMs in OpenShift, with OpenShift Virtualization, Red Hat also supports importing VMs from VMware vSphere, Red Hat Virtualization, and Red Hat OpenStack Platform deployments.



Certain features like live VM migration, VM disk cloning, VM snapshots and so on are also supported by OpenShift Virtualization with assistance from Astra Trident when backed by NetApp ONTAP. Examples of each of these workflows are discussed later in this document in their respective sections.

To learn more about Red Hat OpenShift Virtualization, see the documentation [here](#).

## Deployment for OpenShift Virtualization

### Deploy Red Hat OpenShift Virtualization with NetApp ONTAP

#### Prerequisites

- A Red Hat OpenShift cluster (later than version 4.6) installed on bare-metal infrastructure with RHCOS worker nodes
- The OpenShift cluster must be installed via installer provisioned infrastructure (IPI)
- Deploy Machine Health Checks to maintain HA for VMs

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

Because OpenShift Virtualization is managed by an operator installed on the OpenShift cluster, it imposes additional overhead on memory, CPU, and storage, which must be accounted for while planning the hardware requirements for the cluster. See the documentation [here](#) for more details.

Optionally, you can also specify a subset of the OpenShift cluster nodes to host the OpenShift Virtualization operators, controllers, and VMs by configuring node placement rules. To configure node placement rules for OpenShift Virtualization, follow the documentation [here](#).

For the storage backing OpenShift Virtualization, NetApp recommends having a dedicated StorageClass that requests storage from a particular Trident backend, which in turn is backed by a dedicated SVM. This maintains a level of multitenancy with regard to the data being served for VM-based workloads on the OpenShift cluster.

## Deploy Red Hat OpenShift Virtualization with NetApp ONTAP

To install OpenShift Virtualization, complete the following steps:

1. Log into the Red Hat OpenShift bare-metal cluster with cluster-admin access.
2. Select Administrator from the Perspective drop down.
3. Navigate to Operators > OperatorHub and search for OpenShift Virtualization.



4. Select the OpenShift Virtualization tile and click Install.



## OpenShift Virtualization

2.6.2 provided by Red Hat

[Install](#)

**Latest version**

2.6.2

**Capability level**

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

**Provider type**

Red Hat

**Provider**

Red Hat

### Requirements

Your cluster must be installed on bare metal infrastructure with Red Hat Enterprise Linux CoreOS workers.

### Details

**OpenShift Virtualization** extends Red Hat OpenShift Container Platform, allowing you to host and manage virtualized workloads on the same platform as container-based workloads. From the OpenShift Container Platform web console, you can import a VMware virtual machine from vSphere, create new or clone existing VMs, perform live migrations between nodes, and more. You can use OpenShift Virtualization to manage both Linux and Windows VMs.

The technology behind OpenShift Virtualization is developed in the [KubeVirt](#) open source community. The KubeVirt project extends [Kubernetes](#) by adding additional virtualization resource types through [Custom Resource Definitions](#) (CRDs). Administrators can use Custom Resource Definitions to manage [VirtualMachine](#) resources alongside all other resources that Kubernetes provides.

5. On the Install Operator screen, leave all default parameters and click Install.

Update channel \*

- ☐ 2.1
- ☐ 2.2
- ☐ 2.3
- ☐ 2.4
- ☒ stable

Installation mode \*

- ☐ All namespaces on the cluster (default)  
This mode is not supported by this Operator
- ☒ A specific namespace on the cluster  
Operator will be available in a single Namespace only.

Installed Namespace \*

- ☒ Operator recommended Namespace: **PR** openshift-cnv

**i** Namespace creation

Namespace **openshift-cnv** does not exist and will be created.

- ☐ Select a Namespace

Approval strategy \*

- ☒ Automatic
- ☐ Manual

Install

Cancel

 OpenShift Virtualization  
provided by Red Hat

Provided APIs

**HC** OpenShift  
Virtualization  
Deployment

**Required**

Represents the deployment of  
OpenShift Virtualization

6. Wait for the operator installation to complete.



OpenShift Virtualization  
2.6.2 provided by Red Hat



## Installing Operator

The Operator is being installed. This may take a few minutes.

[View installed Operators in Namespace openshift-cnv](#)

7. After the operator has installed, click Create HyperConverged.



## Installed operator – operand required

The Operator has installed successfully. Create the required custom resource to be able to use this Operator.

**HC** HyperConverged **Required**

Creates and maintains an OpenShift Virtualization Deployment

Create HyperConverged

[View installed Operators in Namespace openshift-cnv](#)

8. On the Create HyperConverged screen, click Create, accepting all default parameters. This step starts the installation of OpenShift Virtualization.

**Name \***

**Labels**

**Infra** >

infra HyperConvergedConfig influences the pod configuration (currently only placement) for all the infra components needed on the virtualization enabled cluster but not necessarily directly on each node running VMs/VMLs.

**Workloads** >

workloads HyperConvergedConfig influences the pod configuration (currently only placement) of components which need to be running on a node where virtualization workloads should be able to run. Changes to Workloads HyperConvergedConfig can be applied only without existing workload.

**Bare Metal Platform**

☒ true

BareMetalPlatform indicates whether the infrastructure is baremetal.

**Feature Gates** >

featureGates is a map of feature gate flags. Setting a flag to `true` will enable the feature. Setting `false` or removing the feature gate, disables the feature.

**Local Storage Class Name**





LocalStorageClassName the name of the local storage class.

- After all the pods move to the Running state in the openshift-cnv namespace and the OpenShift Virtualization operator is in the Succeeded state, the operator is ready to use. VMs can now be created on the OpenShift cluster.

Project: openshift-cnv ▾

## Installed Operators

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

| Name ▾   | Managed Namespaces  | Status  | Last updated  | Provided APIs   |
|--|---|---|---|---|
|  <b>OpenShift Virtualization</b><br>2.6.2 provided by Red Hat |  openshift-cnv |  Succeeded<br>Up to date |  May 18, 8:02 pm | <a href="#">OpenShift Virtualization Deployment</a><br><a href="#">HostPathProvisioner deployment</a> |

## Workflows

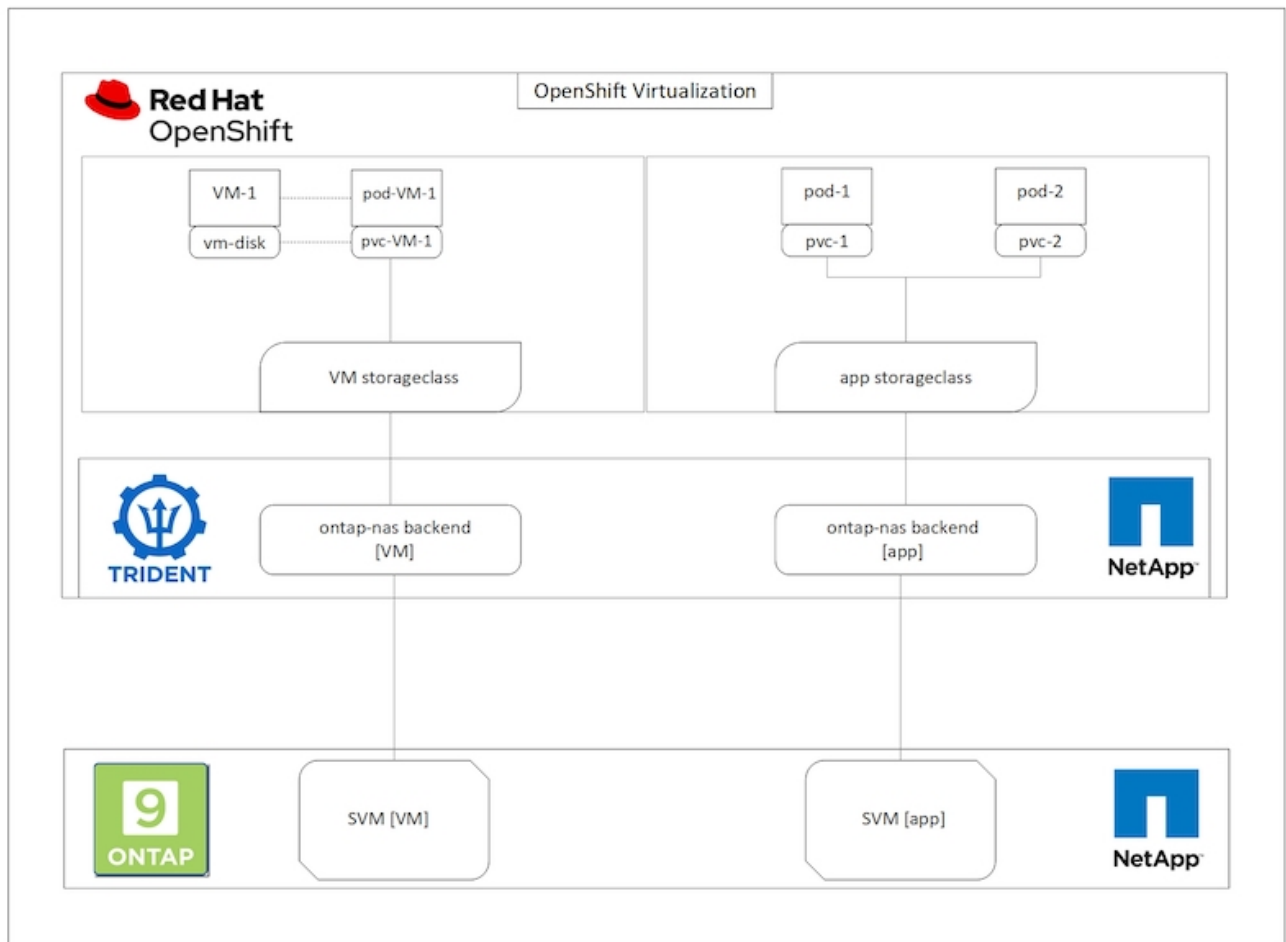
### Workflows: Red Hat OpenShift Virtualization with NetApp ONTAP

#### Create VM

VMs are stateful deployments that require volumes to host the operating system and data. With CNV, because



the VMs are run as pods, the VMs are backed by PVs hosted on NetApp ONTAP through Trident. These volumes are attached as disks and store the entire filesystem including the boot source of the VM.



To create a virtual machine on the OpenShift cluster, complete the following steps:

1. Navigate to Workloads > Virtualization > Virtual Machines and click Create > With Wizard.
2. Select the desired the operating system and click Next.
3. If the selected operating system has no boot source configured, you must configure it. For Boot Source, select whether you want to import the OS image from an URL or from a registry and provide the corresponding details. Expand Advanced and select the Trident-backed StorageClass. Then click Next.

## Boot source

This template does not have a boot source. Provide a custom boot source for this **CentOS 8.0+ VM** virtual machine.

### Boot source type \*

Import via URL (creates PVC) ▼

### Import URL \*

<https://access.cdn.redhat.com/content/origin/files/sha256/58/588167f828001e57688ec4b9b31c11a59d532489f527488ebc89ac5e952...>

Example: For RHEL, visit the [RHEL download page](#) (requires login) and copy the download link URL of the KVM guest image

☒ Mount this as a CD-ROM boot source ?

### Persistent Volume Claim size \*

5 GiB ▼

Ensure your PVC size covers the requirements of the uncompressed image and any other space requirements. More storage can be added later.

### ▼ Advanced

#### Storage class \*

basic (default) ▼

#### Access mode \*

Single User (RWO) ▼

#### Volume mode \*

Filesystem ▼

4. If the selected operating system already has a boot source configured, the previous step can be skipped.
5. In the Review and Create pane, select the project you want to create the VM in and furnish the VM details. Make sure that the boot source is selected to be Clone and boot from CD-ROM with the appropriate PVC assigned for the selected OS.

- 1 Select template
- 2 Review and create

## Review and create

You are creating a virtual machine from the **Red Hat Enterprise Linux 8.0+** VM template.

Project \*

PR default

Virtual Machine Name \* ⓘ

rhel8-light-bat

Flavor \*

Small: 1 CPU | 2 GiB Memory

Storage

40 GiB

Workload profile ⓘ

server

Boot source

Clone and boot from CD-ROM

PVC rhel8

ⓘ A new disk has been added to support the CD-ROM boot source. Edit this disk by customizing the virtual machine.

▼ Disk details

rootdisk-install - Blank - 20GiB - virtio - default Storage class

☒ Start this virtual machine after creation

Create virtual machine

Customize virtual machine

Back

Cancel

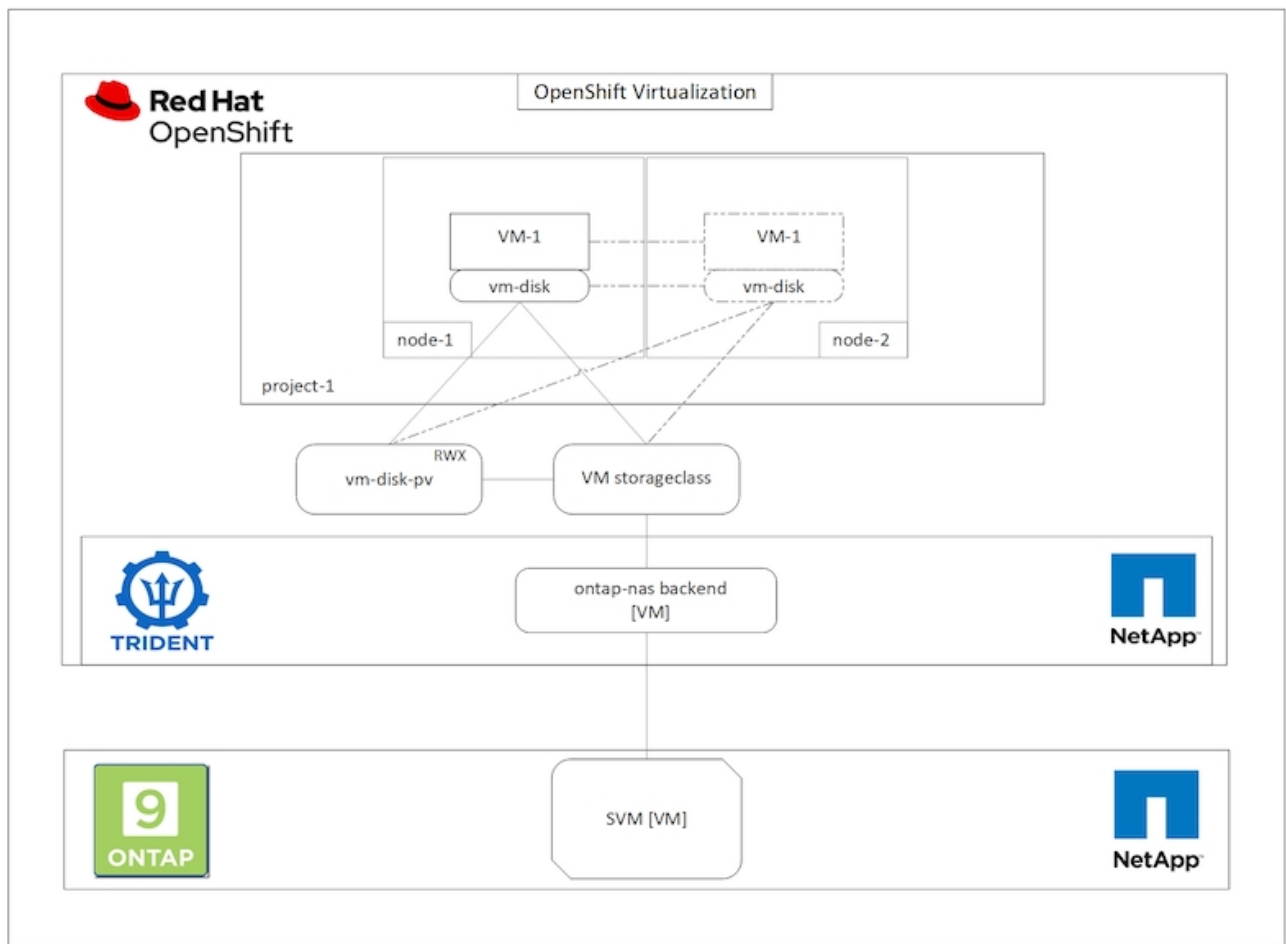
6. If you wish to customize the virtual machine, click **Customize Virtual Machine** and modify the required parameters.
7. Click **Create Virtual Machine** to create the virtual machine; this spins up a corresponding pod in the background.

When a boot source is configured for a template or an operating system from an URL or from a registry, it creates a PVC in the `openshift-virtualization-os-images` project and downloads the KVM guest image to the PVC. You must make sure that template PVCs have enough provisioned space to accommodate the KVM guest image for the corresponding OS. These PVCs are then cloned and attached as rootdisks to virtual machines when they are created using the respective templates in any project.

## Workflows: Red Hat OpenShift Virtualization with NetApp ONTAP

### VM Live Migration

Live Migration is a process of migrating a VM instance from one node to another in an OpenShift cluster with no downtime. For live migration to work in an OpenShift cluster, VMs must be bound to PVCs with shared ReadWriteMany access mode. Astra Trident backend configured with an SVM on a NetApp ONTAP cluster that is enabled for NFS protocol supports shared ReadWriteMany access for PVCs. Therefore, the VMs with PVCs that are requested from StorageClasses provisioned by Trident from NFS-enabled SVM can be migrated with no downtime.



To create a VM bound to PVCs with shared ReadWriteMany access:

1. Navigate to Workloads > Virtualization > Virtual Machines and click Create > With Wizard.
2. Select the desired the operating system and click Next. Let us assume the selected OS already had a boot source configured with it.
3. In the Review and Create pane, select the project you want to create the VM in and furnish the VM details. Make sure that the boot source is selected to be Clone and boot from CD-ROM with the appropriate PVC assigned for the selected OS.
4. Click Customize Virtual Machine and then click Storage.
5. Click the ellipsis next to rootdisk, and make sure that the storageclass provisioned using Trident is selected. Expand Advanced and select Shared Access (RWX) for Access Mode. Then click Save.

## Edit Disk

Type

Disk

Interface \*

virtio

Storage Class

basic (default)

▼ Advanced

Volume Mode

Filesystem

Volume Mode is set by Source PVC

Access Mode

Shared Access (RWX) - Not recommended for basic storage class

 **Access and Volume modes should follow storage feature matrix**  
[Learn more](#) 

Cancel

Save

6. Click Review and confirm and then click Create Virtual Machine.

To manually migrate a VM to another node in the OpenShift cluster, complete the following steps.

1. Navigate to Workloads > Virtualization > Virtual Machines.

2. For the VM you wish to migrate, click the ellipsis, and then click Migrate the Virtual Machine.
3. Click Migrate when the message pops up to confirm.

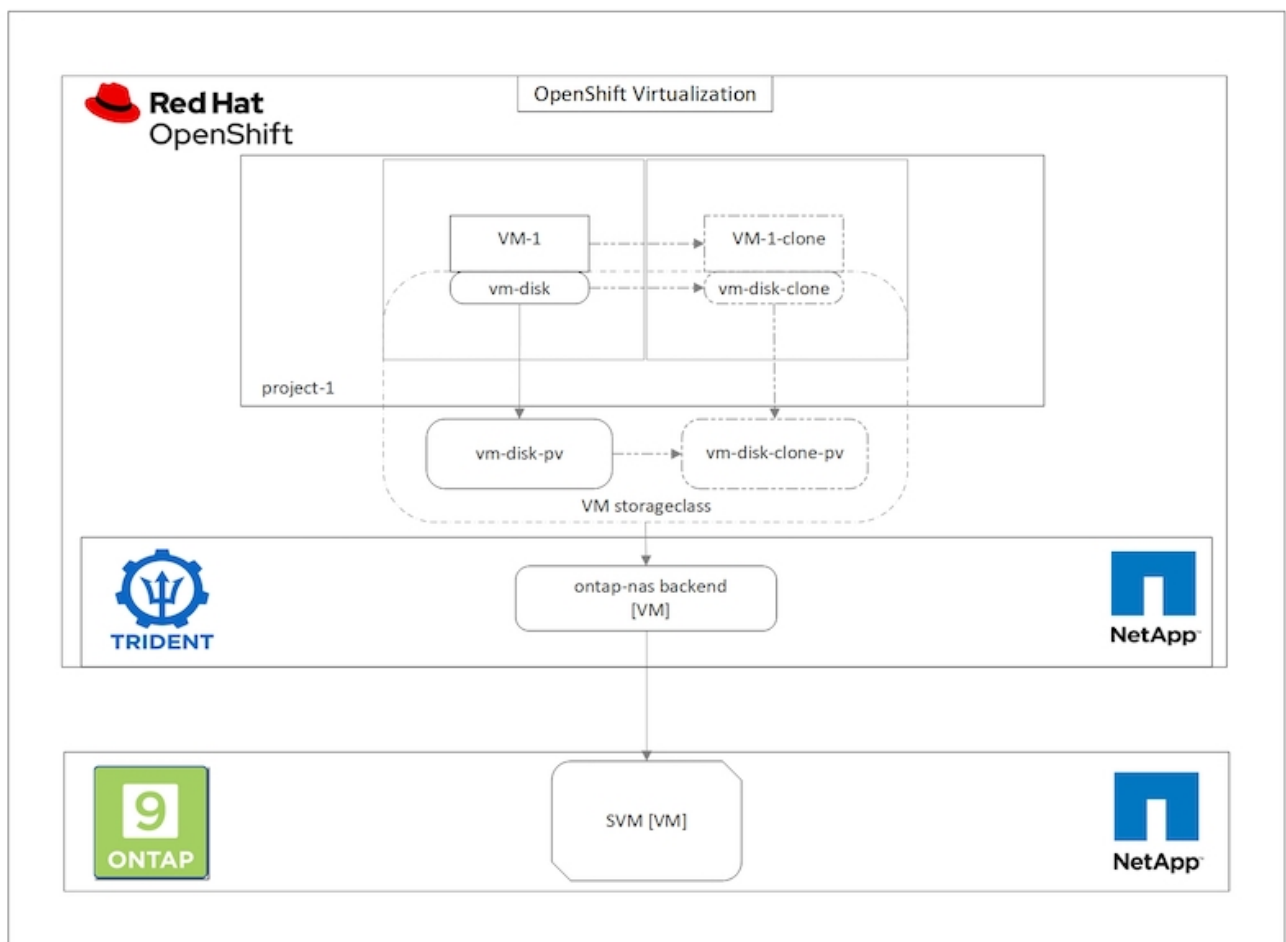


A VM instance in an OpenShift cluster automatically migrates to another node when the original node is placed into maintenance mode if the evictionStrategy is set to LiveMigrate.

## Workflows: Red Hat OpenShift Virtualization with NetApp ONTAP

### VM cloning

Cloning an existing VM in OpenShift is achieved with the support of Astra Trident's Volume CSI cloning feature. CSI volume cloning allows for creation of a new PVC using an existing PVC as the data source by duplicating its PV. After the new PVC is created, it functions as a separate entity and without any link to or dependency on the source PVC.



There are certain restrictions with CSI volume cloning to consider:

1. Source PVC and destination PVC must be in the same project.
2. Cloning is supported within the same storage class.
3. Cloning can be performed only when source and destination volumes use the same VolumeMode setting; for example, a block volume can only be cloned to another block volume.

VMs in an OpenShift cluster can be cloned in two ways:

1. By shutting down the source VM
2. By keeping the source VM live

### **By Shutting down the source VM**

Cloning an existing VM by shutting down the VM is a native OpenShift feature that is implemented with support from Astra Trident. Complete the following steps to clone a VM.

1. Navigate to Workloads > Virtualization > Virtual Machines and click the ellipsis next to the virtual machine you wish to clone.
2. Click Clone Virtual Machine and provide the details for the new VM.

# Clone Virtual Machine

Name \*

rhel8-short-frog-clone

Description

Namespace \*

default



Start virtual machine on clone

Configuration

Operating System

Red Hat Enterprise Linux 8.0 or higher

Flavor

Small: 1 CPU | 2 GiB Memory

Workload Profile

server

NICs

default - virtio

Disks

cloudinitdisk - cloud-init disk

rootdisk - 20Gi - basic



The VM rhel8-short-frog is still running. It will be powered off while cloning.

Cancel

Clone Virtual Machine

3. Click Clone Virtual Machine; this shuts down the source VM and initiates the creation of the clone VM.
4. After this step is completed, you can access and verify the content of the cloned VM.



## By keeping the source VM live

An existing VM can also be cloned by cloning the existing PVC of the source VM and then creating a new VM using the cloned PVC. This method does not require you to shut down the source VM. Complete the following steps to clone a VM without shutting it down.

1. Navigate to Storage > PersistentVolumeClaims and click the ellipsis next to the PVC that is attached to the source VM.
2. Click Clone PVC and furnish the details for the new PVC.

# Clone

Name \*

rhel8-short-frog-rootdisk-28dvv-clone

Access Mode \*

☐ Single User (RWO) ☒ Shared Access (RWX) ☐ Read Only (ROX)

Size \*

20

GiB



PVC details

Namespace

 default

Requested capacity

20 GiB

Access mode

Shared Access (RWX)

Storage Class

 basic

Used capacity

2.2 GiB

Volume mode

Filesystem

Cancel

Clone

3. Then click Clone. This creates a PVC for the new VM.
4. Navigate to Workloads > Virtualization > Virtual Machines and click Create > With YAML.
5. In the spec > template > spec > volumes section, attach the cloned PVC instead of the container disk. Provide all other details for the new VM according to your requirements.

```
- name: rootdisk
  persistentVolumeClaim:
    claimName: rhel8-short-frog-rootdisk-28dvvb-clone
```

6. Click Create to create the new VM.

7. After the VM is created successfully, access and verify that the new VM is a clone of the source VM.

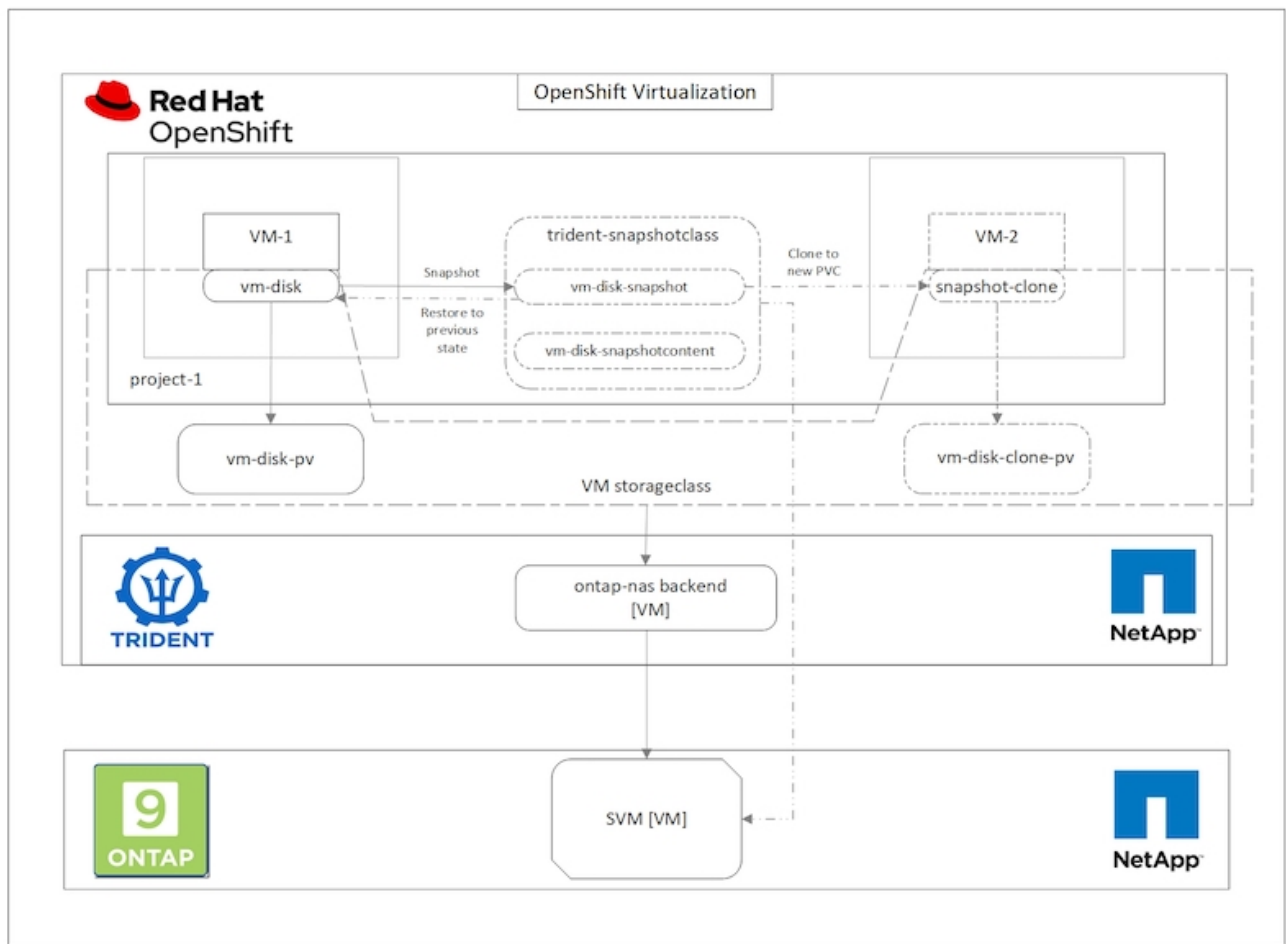
## Workflows: Red Hat OpenShift Virtualization with NetApp ONTAP

### Create VM from a Snapshot

With Astra Trident and Red Hat OpenShift, users can take a snapshot of a persistent volume on Storage Classes provisioned by it. With this feature, users can take a point-in-time copy of a volume and use it to create a new volume or restore the same volume back to a previous state. This enables or supports a variety of use-cases, from rollback to clones to data restore.

For Snapshot operations in OpenShift, the resources `VolumeSnapshotClass`, `VolumeSnapshot`, and `VolumeSnapshotContent` must be defined.

- A `VolumeSnapshotContent` is the actual snapshot taken from a volume in the cluster. It is cluster-wide resource analogous to `PersistentVolume` for storage.
- A `VolumeSnapshot` is a request for creating the snapshot of a volume. It is analogous to a `PersistentVolumeClaim`.
- `VolumeSnapshotClass` lets the administrator specify different attributes for a `VolumeSnapshot`. It allows you to have different attributes for different snapshots taken from the same volume.



To create Snapshot of a VM, complete the following steps:

1. Create a VolumeSnapshotClass that can then be used to create a VolumeSnapshot. Navigate to Storage > VolumeSnapshotClasses and click Create VolumeSnapshotClass.
2. Enter the name of the Snapshot Class, enter `csi.trident.netapp.io` for the driver, and click Create.

```
1  apiVersion: snapshot.storage.k8s.io/v1
2  kind: VolumeSnapshotClass
3  metadata:
4    name: trident-snapshot-class
5  driver: csi.trident.netapp.io
6  deletionPolicy: Delete
7
```

[Create](#)[Cancel](#)[Download](#)

3. Identify the PVC that is attached to the source VM and then create a Snapshot of that PVC. Navigate to Storage > VolumeSnapshots and click Create VolumeSnapshots.
4. Select the PVC that you want to create the Snapshot for, enter the name of the Snapshot or accept the default, and select the appropriate VolumeSnapshotClass. Then click Create.

## Create VolumeSnapshot

[Edit YAML](#)

PersistentVolumeClaim \*

**PVC** rhel8-short-frog-rootdisk-28dvv

Name \*

rhel8-short-frog-rootdisk-28dvv-snapshot

Snapshot Class \*

**VSC** trident-snapshot-class

[Create](#)[Cancel](#)

5. This creates the snapshot of the PVC at that point in time.

### Create a new VM from the snapshot

1. First, restore the Snapshot into a new PVC. Navigate to Storage > VolumeSnapshots, click the ellipsis next to the Snapshot that you wish to restore, and click Restore as new PVC.
2. Enter the details of the new PVC and click Restore. This creates a new PVC.

## Restore as new PVC

When restore action for snapshot **rhel8-short-frog-rootdisk-28dvb-snapshot** is finished a new crash-consistent PVC copy will be created.

Name \*

rhel8-short-frog-rootdisk-28dvb-snapshot-restore

Storage Class \*

 basic

Access Mode \*

☐ Single User (RWO) ☒ Shared Access (RWX) ☐ Read Only (ROX)

Size \*

20

GiB

### VolumeSnapshot details

Created at

 May 21, 12:46 am

Namespace

 default

Status

 Ready

API version

snapshot.storage.k8s.io/v1

Size

20 GiB

3. Next, create a new VM from this PVC. Navigate to Workloads > Virtualization > Virtual Machines and click Create > With YAML.
4. In the spec > template > spec > volumes section, specify the new PVC created from Snapshot instead of

from the container disk. Provide all other details for the new VM according to your requirements.

```
- name: rootdisk
  persistentVolumeClaim:
    claimName: rhel8-short-frog-rootdisk-28dzb-snapshot-restore
```

5. Click Create to create the new VM.
6. After the VM is created successfully, access and verify that the new VM has the same state as that of the VM whose PVC was used to create the snapshot at the time when the snapshot was created.

## Workflows: Red Hat OpenShift Virtualization with NetApp ONTAP

### Migration of VM from VMware to OpenShift Virtualization using Migration Toolkit for Virtualization

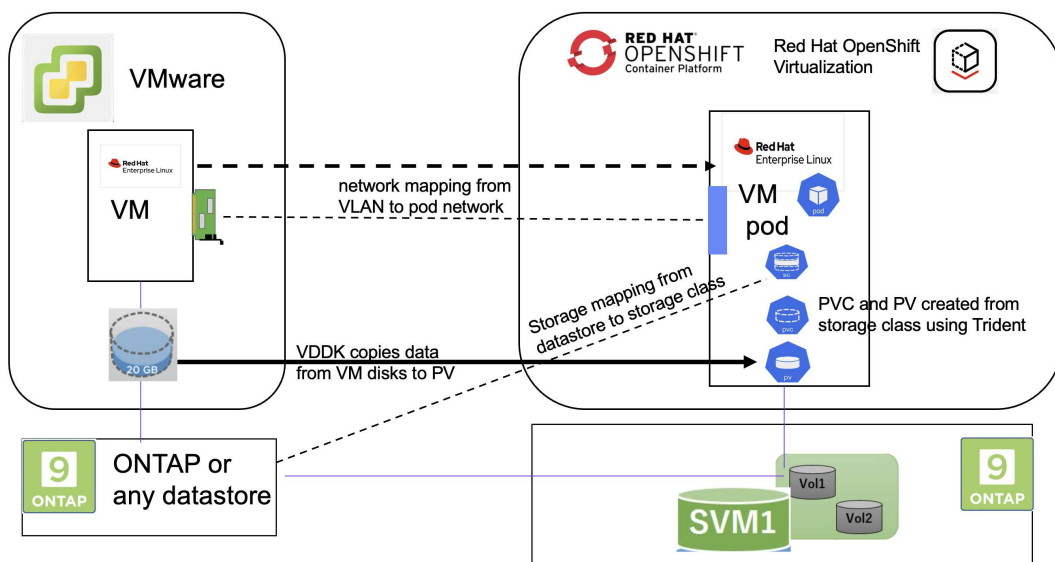
In this section, we will see how to use the Migration Toolkit for Virtualization (MTV) to migrate virtual machines from VMware to OpenShift Virtualization running on OpenShift Container platform and integrated with NetApp ONTAP storage using Astra Trident.

The following video shows a demonstration of the migration of a RHEL VM from VMware to OpenShift Virtualization using ontap-san for persistent storage.

### [Using Red Hat MTV to migrate VMs to OpenShift Virtualization with NetApp ONTAP Storage](#)

The following diagram shows a high level view of the migration of a VM from VMware to Red Hat OpenShift Virtualization.

## Migration of VM from VMware to OpenShift Virtualization



### Prerequisites for the sample migration

## On VMware

- A RHEL 9 VM using rhel 9.3 with the following configurations were installed:
  - CPU: 2, Memory: 20 GB, Hard disk: 20 GB
  - user credentials: root user and an admin user credentials
- After the VM was ready, postgresql server was installed.
  - postgresql server was started and enabled to start on boot

```
systemctl start postgresql.service`  
systemctl enable postgresql.service  
The above command ensures that the server can start in the VM in  
OpenShift Virtualization after migration
```

- Added 2 databases, 1 table and 1 row in the table were added. Refer [here](#) for the instructions for installing postgresql server on RHEL and creating database and table entries.



Ensure that you start the postgresql server and enable the service to start at boot.

## On OpenShift Cluster

The following installations were completed before installing MTV:

- OpenShift Cluster 4.13.34
- [Astra Trident 23.10](#)
- Multipath on the cluster nodes enabled for iSCSI (for ontap-san storage class). See the provided yaml to create a daemon set that enables iSCSI on each node in the cluster.
- Trident backend and Storage class for ontap SAN using iSCSI. See the provided yaml files for trident backend and storage class.
- [OpenShift Virtualization](#)

To install iscsi and multipath on the OpenShift Cluster nodes use the yaml file given below

### Preparing the cluster nodes for iSCSI

```
apiVersion: apps/v1  
kind: DaemonSet  
metadata:  
  namespace: trident  
  name: trident-iscsi-init  
  labels:  
    name: trident-iscsi-init  
spec:  
  selector:  
    matchLabels:  
      name: trident-iscsi-init  
  template:
```

```

metadata:
  labels:
    name: trident-iscsi-init
spec:
  hostNetwork: true
  serviceAccount: trident-node-linux
  initContainers:
  - name: init-node
    command:
      - nsenter
      - --mount=/proc/1/ns/mnt
      - --
      - sh
      - -c
    args: ["$(STARTUP_SCRIPT)"]
    image: alpine:3.7
    env:
    - name: STARTUP_SCRIPT
      value: |
        #! /bin/bash
        sudo yum install -y lsscsi iscsi-initiator-utils sg3_utils
device-mapper-multipath
    rpm -q iscsi-initiator-utils
    sudo sed -i 's/^\(node.session.scan\) .*/\1 = manual/'
/etc/iscsi/iscsid.conf
    cat /etc/iscsi/initiatorname.iscsi
    sudo mpathconf --enable --with_multipathd y --find_multipaths
n
    sudo systemctl enable --now iscsid multipathd
    sudo systemctl enable --now iscsi
  securityContext:
    privileged: true
  hostPID: true
  containers:
  - name: wait
    image: k8s.gcr.io/pause:3.1
  hostPID: true
  hostNetwork: true
  tolerations:
  - effect: NoSchedule
    key: node-role.kubernetes.io/master
updateStrategy:
  type: RollingUpdate

```

Use the following yaml file to create trident backend configuration for using ontap san storage  
**Trident backend for iSCSI**



```

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

```

Use the following yaml file to create trident storage class configuration for using ontap san storage  
**Trident storage class for iSCSI**

```

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

```

## Install MTV

Now you can install the Migration Toolkit for virtualization (MTV). Refer to the instructions provided [here](#) for help with the installation.

The Migration Toolkit for Virtualization (MTV) user interface is integrated into the OpenShift web console. You can refer [here](#) to start using the user interface for various tasks.

## Create Source Provider

In order to migrate the RHEL VM from VMware to OpenShift Virtualization, you need to first create the source provider for VMware. Refer to the instructions [here](#) to create the source provider.

You need the following to create your VMware source provider:

- VCenter url
- VCenter Credentials
- VCenter server thumbprint
- VDDK image in a repository

Sample source provider creation:

Select provider type \*

vm vSphere

Provider resource name \*

vmware-source

Unique Kubernetes resource name identifier

URL \*

URL of the vCenter SDK endpoint. Ensure the URL includes the "/sdk" path. For example: https://vCenter-host-example.com/sdk

VDDK init image

docker.repo.eng.netapp.com/banum/vddk:801

VDDK container image of the provider, when left empty some functionality will not be available

Username \*

administrator@vsphere.local

vSphere REST API user name.

Password \*

.....

vSphere REST API password credentials.

SSHA-1 fingerprint \*

The provider currently requires the SHA-1 fingerprint of the vCenter Server's TLS certificate in all circumstances. vSphere calls this the server's thumbprint.

Skip certificate validation

☒



The Migration Toolkit for Virtualization (MTV) uses the VMware Virtual Disk Development Kit (VDDK) SDK to accelerate transferring virtual disks from VMware vSphere. Therefore, creating a VDDK image, although optional, is highly recommended. To make use of this feature, you download the VMware Virtual Disk Development Kit (VDDK), build a VDDK image, and push the VDDK image to your image registry.

Follow the instructions provided [here](#) to create and push the VDDK image to a registry accessible from the OpenShift Cluster.

## Create Destination provider

The host cluster is automatically added as the OpenShift virtualization provider is the source provider.

## Create Migration Plan

Follow the instructions provided [here](#) to create a migration plan.

While creating a plan, you need to create the following if not already created:

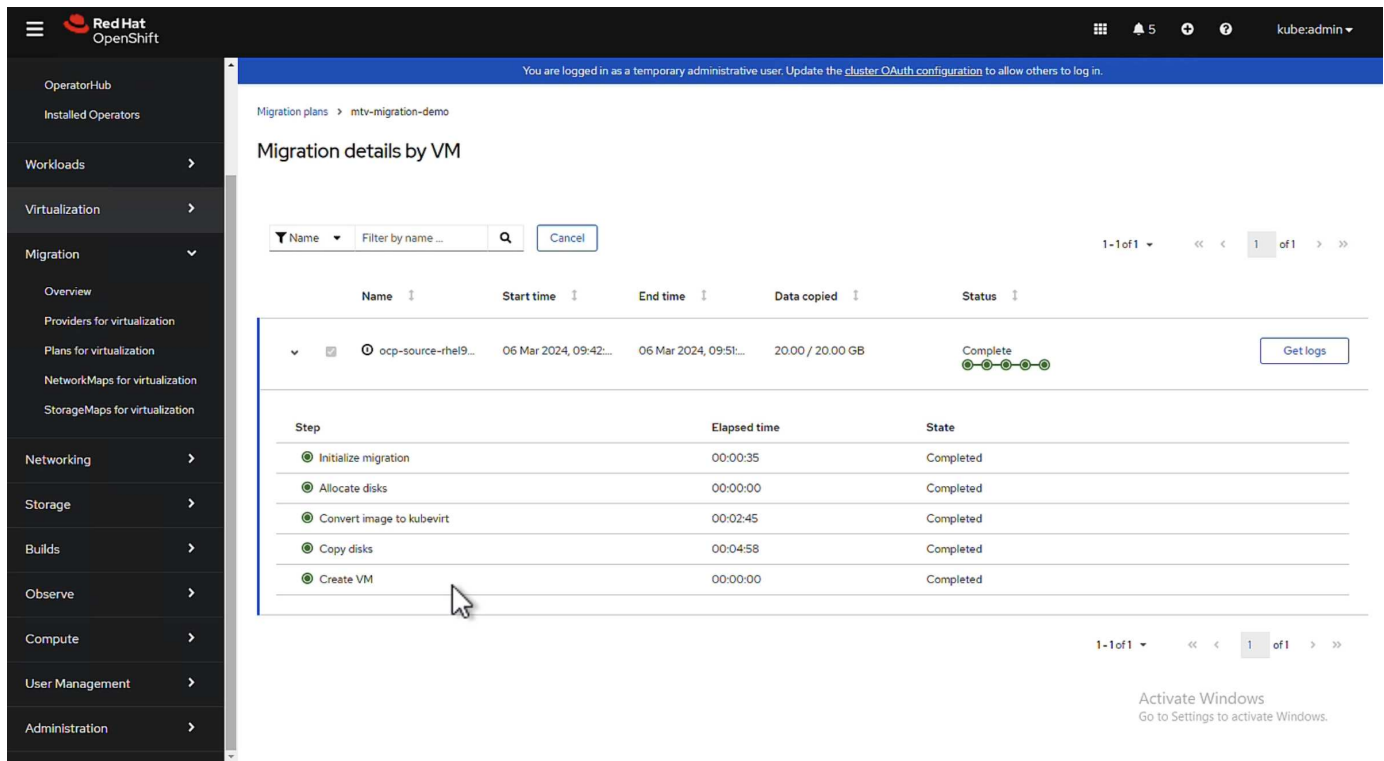
- A network mapping to map the source network to the target network.
- A storage mapping to map the source datastore to the target storage class. For this you can choose ontap-san storage class.

Once the migration plan is created, the status of the plan should show **Ready** and you should now be able to **Start** the plan.

The screenshot shows the Red Hat OpenShift MTV interface. The left sidebar contains the navigation menu with 'Migration' expanded. The main panel displays a table of migration plans. The first plan, 'mtv-migration-demo', is in 'Ready' status and has a 'Start' button. A hand cursor is pointing at the 'Start' button. Other plans show 'Succeeded' status with '1 of 1 VMs migrated'.

| Name                       | Source  | Target | VMs | Status    | Description                                |
|----------------------------|---------|--------|-----|-----------|--|
| mtv-migration-demo         | vmware  | host   | 1   | Ready     | Plan for migrating VM to OpenShift Virt... |
| vmware-osv-migration       | vmware2 | host   | 1   | Succeeded | Migrating RHEL 9 vm to OpenShift Virtu...  |
| vmware-osv-migration-plan1 | vmware2 | host   | 1   | Succeeded | 1 of 1 VMs migrated                        |
| vmware-osv-migration-plan2 | vmware2 | host   | 1   | Succeeded | 1 of 1 VMs migrated                        |

Clicking on **Start** will run through a sequence of steps to complete the migration of the VM.



When all steps are completed, you can see the migrated VMs by clicking on the **virtual machines** under **Virtualization** in the left-side navigation menu. Instructions to access the virtual machines are provided [here](#).

You can log into the virtual machine and verify the contents of the postgresql databases. The databases, tables and the entries in the table should be the same as what was created on the source VM.

## Data Protection for OpenShift Virtualization

### Data protection for VMs in OpenShift Virtualization using OpenShift API for Data Protection (OADP)

Author: Banu Sundhar, NetApp

This section of the reference document provides details for creating backups of VMs using the OpenShift API for Data Protection (OADP) with Velero on NetApp ONTAP S3 or NetApp StorageGRID S3. The backups of Persistent Volumes(PVs) of the VM disks are created using CSI Astra Trident Snapshots.

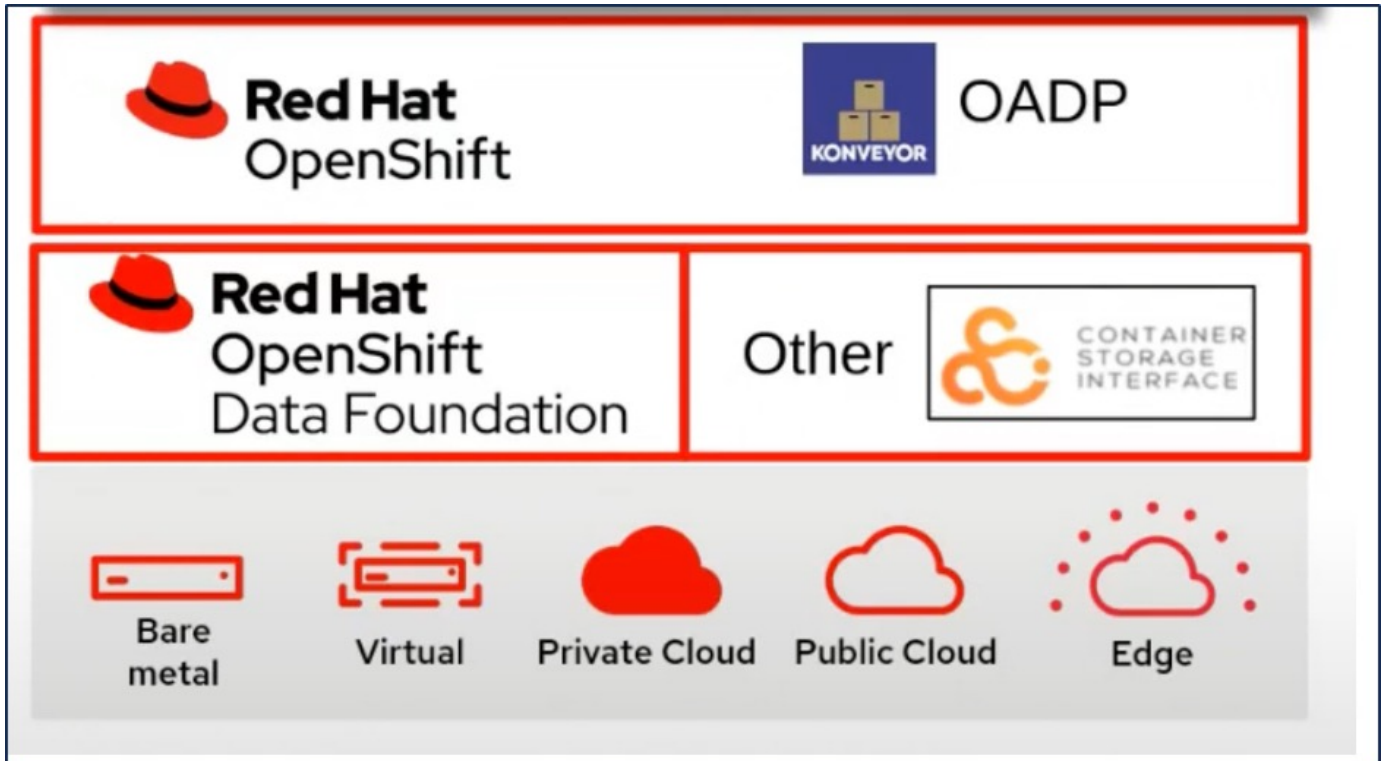
Virtual machines in the OpenShift Virtualization environment are containerized applications that run in the worker nodes of your OpenShift Container platform. It is important to protect the VM metadata as well as the persistent disks of the VMs, so that when they are lost or corrupted, you can recover them.

The persistent disks of the OpenShift Virtualization VMs 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 VMs including its data volumes to

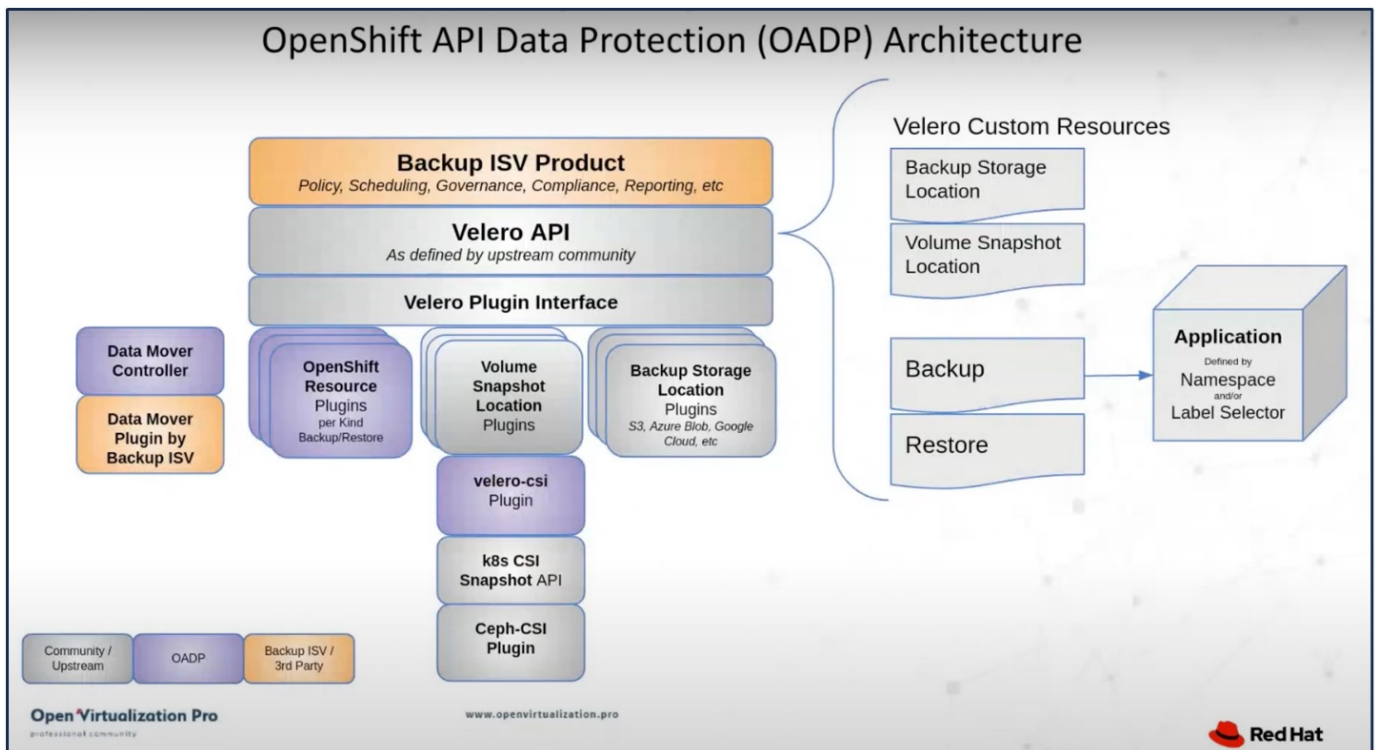
- ONTAP Object Storage
- StorageGrid

We then restore from the backup when needed.

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
- OpenShift Virtualization installed via OperatorOpenShift Virtualization Operator provided by Red Hat
- OADP Operator 1.13 provided by Red Hat
- Velero CLI 1.13 for Linux
- Astra Trident 24.02
- ONTAP 9.12

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

## 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

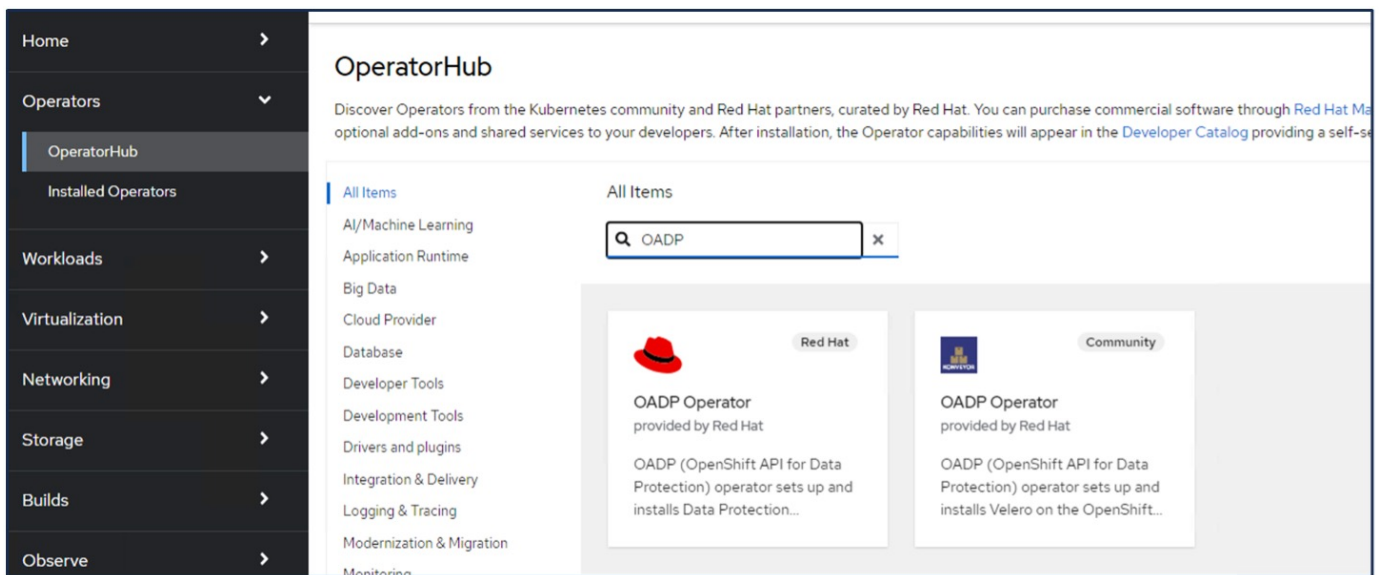
- Trident Snapshot class created on the cluster
- Cluster-admin access to Red Hat OpenShift cluster
- Admin access to NetApp ONTAP cluster
- OpenShift Virtualization operator installed and configured
- VMs deployed in a Namespace on OpenShift Virtualization
- An admin workstation with tridentctl and oc tools installed and added to \$PATH



If you want to take a backup of a VM when it is in the Running state, then you must install the QEMU guest agent on that virtual machine. If you install the VM using an existing template, then QEMU agent is installed automatically. QEMU allows the guest agent to quiesce in-flight data in the guest OS during the snapshot process, and avoid possible data corruption. If you do not have QEMU installed, you can stop the virtual machine before taking a backup.

## 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><br>4.14.4 provided by Red Hat |  openshift-cnv                            |  openshift-cnv                            |  Succeeded<br>Up to date |
|  <b>OADP Operator</b><br>1.3.0 provided by Red Hat             |  openshift-adp                            |  openshift-adp                            |  Succeeded<br>Up to date |
|  <b>Package Server</b><br>0.0.1-snapshot provided by           |  openshift-operator-lifecycle-<br>manager |  openshift-operator-lifecycle-<br>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.

The screenshot displays the OpenShift web console interface. On the left, a sidebar contains navigation links: 'Installed Operators', 'Workloads' (expanded), 'Pods', 'Deployments', 'DeploymentConfigs', 'StatefulSets', 'Secrets' (selected), and 'ConfigMaps'. The main content area is titled 'Secrets' and shows a table of existing secrets. The table has columns for 'Name', 'Type', 'Size', and 'Created'. Two secrets are listed: 'builder-dockercfg-7g8ww' (Type: kubernetes.io/dockercfg, Size: 1) and 'builder-token-rm4s' (Type: kubernetes.io/service-account-token, Size: 4). Both were created on April 11, 2024, at 10:52 AM. In the top right corner, there is a 'Create' button with a dropdown menu open, showing options: 'Key/value secret', 'Image pull secret', 'Source secret', 'Webhook secret', and 'From YAML'.

| Name                    | Type                                | Size | Created                |
|-------------------------|-------------------------------------|------|------------------------|
| builder-dockercfg-7g8ww | kubernetes.io/dockercfg             | 1    | Apr 11, 2024, 10:52 AM |
| builder-token-rm4s      | kubernetes.io/service-account-token | 4    | Apr 11, 2024, 10:52 AM |

Project: openshift-adp ▼

## Create key/value secret

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

**Secret name \***

Unique name of the new secret.

**Key \***

**Value**

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

```
[default]
aws_access_key_id=<Access Key Id of S3 user>
aws_secret_access_key=<Secret Access Key of S3 user>
```

+ Add key/value





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

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

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         | <div> <div>Name</div> <div>Search by name...</div> </div>   |  |  |   |  |
| Installed Operators |   |  |  |   |  |
| Workloads           |   |  |  |   |  |
| Virtualization      |   |  |  |   |  |
| Networking          |   |  |  |   |  |
|                     | <div>Name</div> <div></div>   | <div>Managed Namespaces</div> <div></div>  | <div>Status</div> <div></div>  | <div>Last updated</div> <div></div>   | <div>Provided APIs</div> <div></div>   |
|                     | <div> OADP Operator</div> <div>1.3.0 provided by Red Hat</div>   | <div> openshift-adp</div> | <div> Succeeded</div> <div>Up to date</div> | <div> Apr 11, 2024, 10:53 AM</div> | <div><a href="#">BackupRepository</a></div> <div><a href="#">Backup</a></div> <div><a href="#">BackupStorageLocation</a></div> <div><a href="#">DeleteBackupRequest</a></div> <div><a href="#">View 11 more...</a></div> |

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: 'true' ->use this for https communication
with ONTAP S3
            profile: default
            region: us-east
            s3ForcePathStyle: 'True' ->This allows use of IP in s3URL
            s3Url: 'https://10.xx.xx.xx' ->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: demobackup ->The folder that will be created in the
bucket
            provider: aws
        configuration:
            nodeAgent:
                enable: true
                uploaderType: kopia
                #default Data Mover uses Kopia to move snapshots to Object Storage
            velero:
                defaultPlugins:
                    - csi ->Add this plugin
                    - openshift
                    - aws
                    - kubevirt ->Add this plugin

```

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

```
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

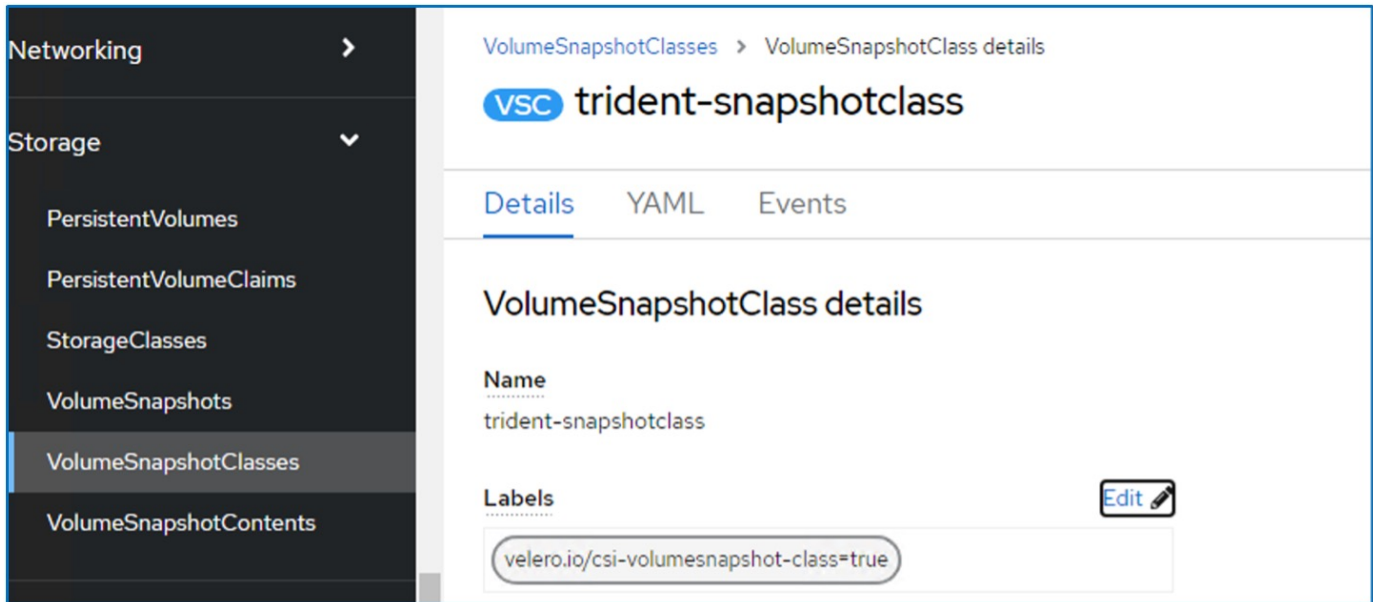
If you use Container Storage Interface (CSI) snapshots, you do not need to specify a snapshot location because you will create a VolumeSnapshotClass CR to register the CSI driver. In our example, you use Astra Trident CSI and you have previously created VolumeSnapShotClass CR using the Trident CSI driver.

### 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
```

VolumeSnapshotClasses > VolumeSnapshotClass details

## VSC trident-snapshotclass

Details | YAML | Events

### VolumeSnapshotClass details

**Name**  
trident-snapshotclass

**Labels** [Edit](#)

velero.io/csi-volumesnapshot-class=true


**Annotations**  
[1 annotation](#)

**Driver**  
csi.trident.netapp.io

**Deletion policy**  
Retain

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

Installed Operators > Operator details

 **OADP Operator**  
1.3.0 provided by Red Hat

[Actions](#)

[ServerStatusRequest](#) [VolumeSnapshotLocation](#) [DataDownload](#) [DataUpload](#) [CloudStorage](#) [DataProtectionApplication](#)

### DataProtectionApplications

[Create DataProtectionApplication](#)


Name [Search by name...](#)

| Name                            | Kind                      | Status                | Labels    |
|---------------------------------|---------------------------|-----------------------|-----------|
| <a href="#">DPA velero-demo</a> | DataProtectionApplication | Condition: Reconciled | No labels |

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**  
1.3.0 provided by Red Hat


Actions ▾

Repository Backup BackupStorageLocation DeleteBackupRequest DownloadRequest PodVolumeBackup PodVolumeRe

## BackupStorageLocations

Create BackupStorageLocation

Name ▾ Search by name... /

| Name ▴ ▾   | Kind ▴ ▾              | Status ▴ ▾       | Labels ▴ ▾   |
|--|-----------------------|------------------|--|
|  <b>velero-demo-1</b> | BackupStorageLocation | Phase: Available | <div>app.kubernetes.io/component=bsl</div> <div>app.kubernetes.io/instance=velero-demo-1</div> <div>app.kubernetes.io/managed-by=oadp-oper...</div> <div>app.kubernetes.io/name=oadp-operator-ve...</div> <div>openshift.io/oadp=True</div> <div>openshift.io/oadp-registry=True</div> |

## Creating on-demand backup for VMs in OpenShift Virtualization

### Steps to create a backup of a VM

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

A snapshot of the persistent volumes backing the disks will be created by the CSI. A backup of the VM along with the snapshot of its disks are created and stored in the backup location specified in the yaml. The backup will remain in the system for 30 days as specified in the ttl.

```
apiVersion: velero.io/v1
kind: Backup
metadata:
  name: backup1
  namespace: openshift-adp
spec:
  includedNamespaces:
  - virtual-machines-demo
  snapshotVolumes: true
  storageLocation: velero-demo-1 -->this is the backupStorageLocation
  previously created
  ttl: 720h0m0s
```


when Velero is configured.



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

Project: openshift-adp

Installed Operators > Operator details

 OADP Operator

1.3.0 provided by Red Hat

Actions

Details

YAML

Subscription

Events

All instances

BackupRepository

Backup

BackupStorageLocation

DeleteBa

Backups

Create Backup

Name

Search by name...

| Name    | Kind   | Status                        | Labels                                   |
|---------|--------|-------------------------------|--|
| backup1 | Backup | Phase: <span>Completed</span> | velero.io/storage-location=velero-demo-1 |

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



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

Path: / demobackup/ backups/ backup1/

| Name                                       | Size      | Type      | Last Modified         | Storage Class |
|--|-----------|-----------|-----------------------|---------------|
| ..   |           |           |                       |               |
| backup1.tar.gz                             | 230.36 KB | GZ File   | 4/15/2024 10:26:29 PM | STANDARD      |
| velero-backup.json                         | 3.35 KB   | JSON File | 4/15/2024 10:26:29 PM | STANDARD      |
| backup1-resource-list.json.gz              | 1.12 KB   | GZ File   | 4/15/2024 10:26:29 PM | STANDARD      |
| backup1-itemoperations.json.gz             | 600 bytes | GZ File   | 4/15/2024 10:26:28 PM | STANDARD      |
| backup1-volumesnapshots.json.gz            | 29 bytes  | GZ File   | 4/15/2024 10:26:28 PM | STANDARD      |
| backup1-podvolumebackups.json.gz           | 29 bytes  | GZ File   | 4/15/2024 10:26:28 PM | STANDARD      |
| backup1-results.gz                         | 49 bytes  | GZ File   | 4/15/2024 10:26:28 PM | STANDARD      |
| backup1-csi-volumesnapshotclasses.json.gz  | 426 bytes | GZ File   | 4/15/2024 10:26:28 PM | STANDARD      |
| backup1-csi-volumesnapshotcontents.json.gz | 1.43 KB   | GZ File   | 4/15/2024 10:26:28 PM | STANDARD      |
| backup1-csi-volumesnapshots.json.gz        | 1.34 KB   | GZ File   | 4/15/2024 10:26:28 PM | STANDARD      |
| backup1-logs.gz                            | 13.49 KB  | GZ File   | 4/15/2024 10:26:28 PM | STANDARD      |

## Creating scheduled backups for VMs in OpenShift Virtualization

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.

```

apiVersion: velero.io/v1
kind: Schedule
metadata:
  name: <schedule>
  namespace: openshift-adp
spec:
  schedule: 0 7 * * *
  template:
    hooks: {}
    includedNamespaces:
    - <namespace>
    storageLocation: velero-demo-1
    defaultVolumesToFsBackup: true
    ttl: 720h0m0s

```


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.

Project: openshift-adp ▼

---

[Installed Operators](#) > [Operator details](#)

 **OADP Operator**  
1.3.0 provided by Red Hat



---

[storageLocation](#) [DeleteBackupRequest](#) [DownloadRequest](#) [PodVolumeBackup](#) [PodVolumeRestore](#) [Restore](#) [Schedule](#)

---

## Schedules


Name ▼ Search by name... /

| Name ↑  | Kind ↑   | Status ↑   | Labels ↑  |
|---|----------|--|-----------|
|  schedule1 | Schedule | Phase:  Enabled | No labels |

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

1.3.0 provided by Red Hat

Actions

Events

All instances

BackupRepository

Backup

BackupStorageLocation

DeleteBackupRequest


DownloadRequest

Backups

Create Backup

Name

Search by name...

| Name   | Kind   | Status            | Labels   |
|--|--------|-------------------|--|
|  schedule1-20240416140507 | Backup | Phase: InProgress | <div>velero.io/schedule-name=schedule1</div> <div>velero.io/storage-location=velero-demo-1</div> |

## Restore a VM from a backup

### Prerequisites


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

## 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

DownloadRequest

PodVolumeBackup

PodVolumeRestore

**Restore**

Schedule

ServerStatusRequest

VolumeSnap

Restores


Create Restore

```
apiVersion: velero.io/v1
kind: Restore
metadata:
  name: restore1
  namespace: openshift-adp
spec:
  backupName: backup1
  restorePVs: true
```

When the phase shows completed, you can see that the virtual machines have been restored to the state when the snapshot was taken. (If the backup was created when the VM was running, restoring the VM from the backup will start the restored VM and bring it to a running state). The VM is restored to the same namespace.

Project: openshift-adp

Installed Operators > Operator details

 **OADP Operator**  
1.3.0 provided by Red Hat

Actions

DownloadRequest

PodVolumeBackup

PodVolumeRestore

**Restore**

Schedule

ServerStatusRequest



VolumeSn

Restores

Create Restore

Name

Search by name...

| Name   | Kind    | Status   | Labels    |
|--|---------|--|-----------|
|  restore1 | Restore | Phase:  Completed | No labels |

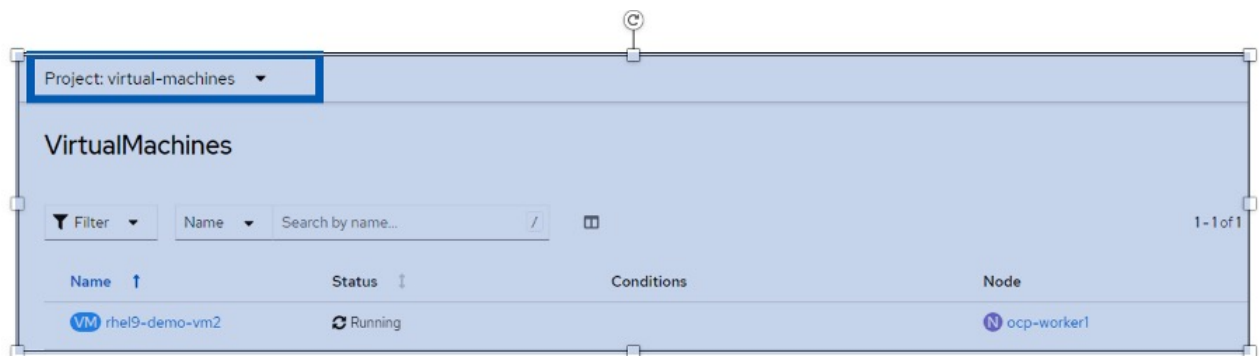
## Restore to a different namespace

To restore the VM 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 a VM and its disks in the virtual-machines-demo namespace when the backup was taken to the virtual-machines namespace.

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

When the phase shows completed, you can see that the virtual machines have been restored to the state when the snapshot was taken. (If the backup was created when the VM was running, restoring the VM from the backup will start the restored VM and bring it to a running state). The VM is restored to a different namespace as specified in the yaml.



## 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 virtual machine, during creation uses ontap-nas as the storage class for its disks. A backup of the virtual machine named backup1 is created.

The screenshot shows the 'VirtualMachine details' page for 'rhel9-demo-vm1' in the 'virtual-machines-demo' project. The 'Configuration' tab is selected, showing a table of disks. The table has columns: Name, Source, Size, Drive, Interface, and Storage class. There are three disks listed: 'cloudinitdisk' (Source: Other, Size: -, Drive: Disk, Interface: virtio, Storage class: -), 'disk1' (Source: PVC rhel9-demo-vm1-disk1, Size: 31.75 GiB, Drive: Disk, Interface: virtio, Storage class: ontap-nas), and 'rootdisk' (Source: PVC rhel9-demo-vm1, Size: 31.75 GiB, Drive: Disk, Interface: virtio, Storage class: ontap-nas). The 'rootdisk' is marked as 'bootable'.

| Name          | Source                   | Size      | Drive | Interface | Storage class |
|---------------|--------------------------|-----------|-------|-----------|---------------|
| cloudinitdisk | Other                    | -         | Disk  | virtio    | -             |
| disk1         | PVC rhel9-demo-vm1-disk1 | 31.75 GiB | Disk  | virtio    | ontap-nas     |
| rootdisk      | PVC rhel9-demo-vm1       | 31.75 GiB | Disk  | virtio    | ontap-nas     |

The screenshot shows the 'Backup' page for the 'OADP Operator' in the 'openshift-adp' project. The 'Backup' tab is selected, showing a table of backups. The table has columns: Name, Kind, and Status. There is one backup listed: 'backup1' (Kind: Backup, Status: Phase: Completed). A 'Create Backup' button is visible in the top right corner.

| Name    | Kind   | Status           |
|---------|--------|------------------|
| backup1 | Backup | Phase: Completed |

Simulate a loss of the VM by deleting the VM.

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-storage-class-config (can be any name)

Key: change-storage-class-config.yaml:

Value:

```
version: v1
resourceModifierRules:
- conditions:
    groupResource: persistentvolumeclaims
    resourceNameRegex: "^rhel*"
    namespaces:
    - virtual-machines-demo
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-storage-class-config

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

**Key \***

change-storage-class-config.yaml

**Value**

Browse...

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

```
version: v1
resourceModifierRules:
- conditions:
    groupResource: persistentvolumeclaims
```

[+ Add key/value](#) [- Remove key/value](#)

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

```
# kubectl describe cm/change-storage-class-config -n openshift-
adp
Name:          change-storage-class-config
Namespace:     openshift-adp
Labels:        velero.io/change-storage-class=RestoreItemAction
                velero.io/plugin-config=
Annotations:   <none>

Data
====
change-storage-class-config.yaml:
----
version: v1
resourceModifierRules:
- conditions:
    groupResource: persistentvolumeclaims
    resourceNameRegex: "^rhel*"
    namespaces:
    - virtual-machines-demo
  patches:
  - operation: replace
    path: "/spec/storageClassName"
    value: "ontap-nas-eco"

BinaryData
====

Events:   <none>
```

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.

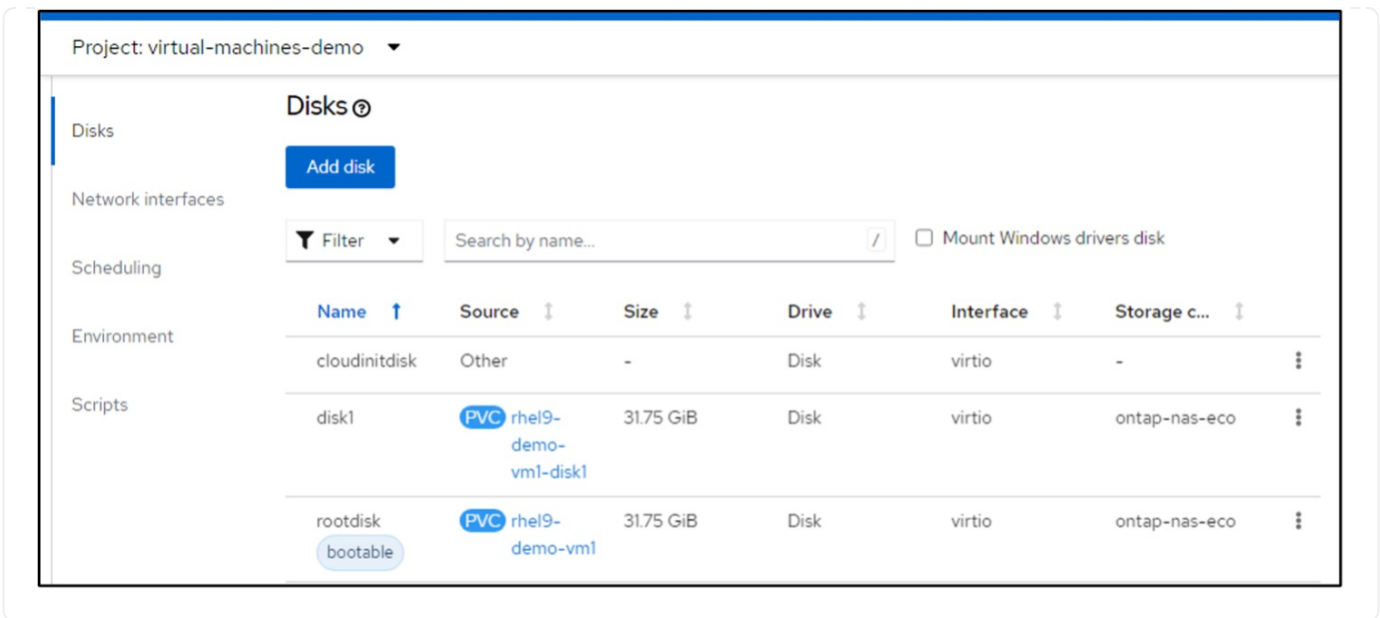
## Step 2

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 VM is restored in the same namespace with the disks created using the storage class ontap-nas-eco.





## Deleting backups and restores in using Velero

### Deleting a backup

You can delete a Backup CR without deleting the Object Storage data by using the OC CLI tool.

```
oc delete backup <backup_CR_name> -n <velero_namespace>
```

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

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

Execute the following delete command using the Velero CLI

```
velero backup delete <backup_CR_name> -n <velero_namespace>
```

You can also delete the Restore CR using the Velero CLI

```
velero restore delete restore --namespace openshift-adp
```

You can use oc command as well as the UI to delete the restore CR

```
oc delete backup <backup_CR_name> -n <velero_namespace>
```

# Monitoring using Cloud Insights

## Monitoring using Cloud Insights for VMs in Red Hat OpenShift Virtualization

Author: Banu Sundhar, NetApp

This section of the reference document provides details for integrating NetApp Cloud Insights with a Red Hat OpenShift Cluster to monitor OpenShift Virtualization VMs.

NetApp Cloud Insights is a cloud infrastructure monitoring tool that gives you visibility into your complete infrastructure. With Cloud Insights, you can monitor, troubleshoot, and optimize all your resources including your public clouds and your private data centers. For more information about NetApp Cloud Insights, refer to the [Cloud Insights documentation](#).

To start using Cloud Insights, you must sign up on the NetApp BlueXP portal. For details, refer to the [Cloud Insights Onboarding](#)

Cloud Insights has several features that enable you to quickly and easily find data, troubleshoot issues, and provide insights into your environment. You can find data easily with powerful queries, you can visualize data in dashboards, and send email alerts for data thresholds you set. Refer to the [video tutorials](#) to help you understand these features.

For Cloud Insights to start collecting data you need the following

### Data Collectors

There are 3 types of Data Collectors:

- \* Infrastructure (storage devices, network switches, compute infrastructure)
- \* Operating Systems (such as VMware or Windows)
- \* Services (such as Kafka)

Data Collectors discover information from the data sources, such as ONTAP storage device (infrastructure data collector). The information gathered is used for analysis, validation, monitoring, and troubleshooting.

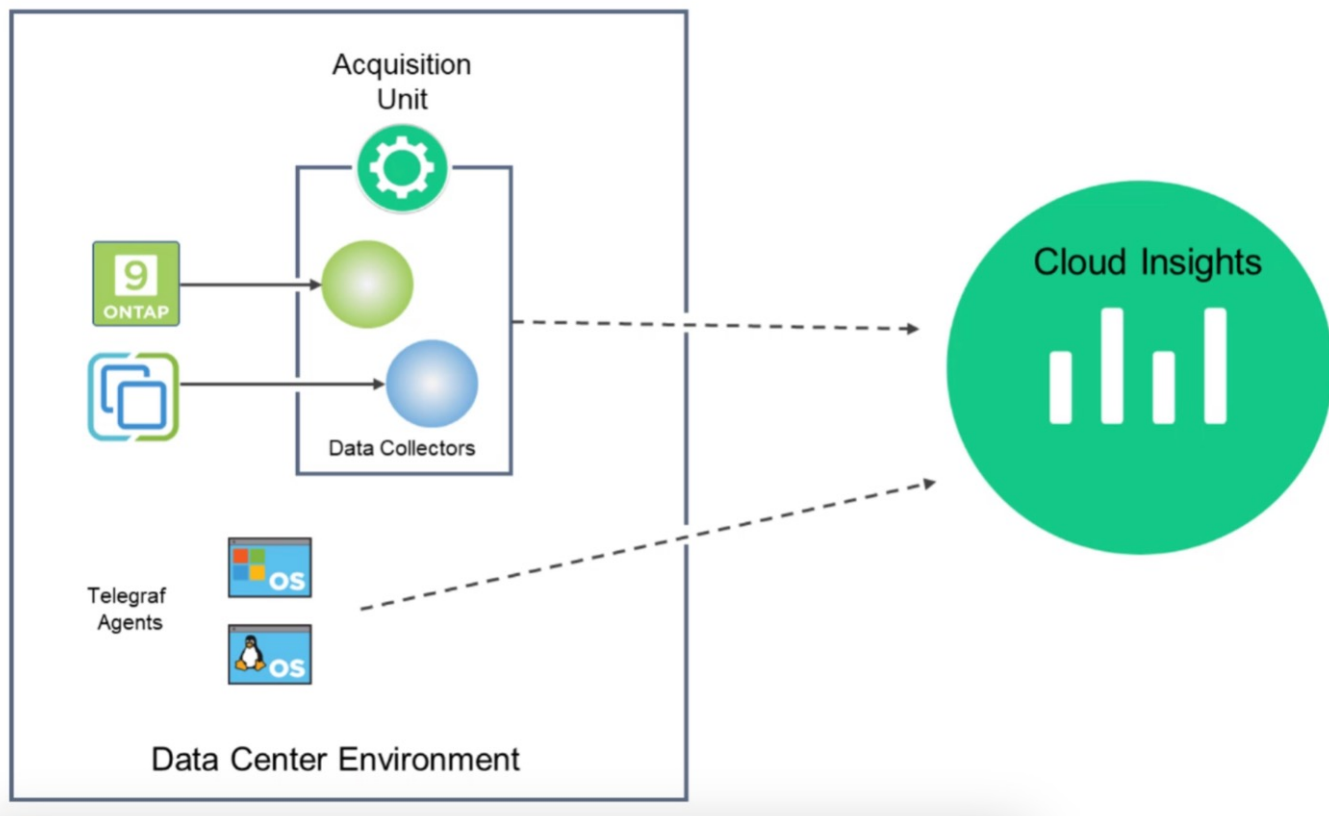
### Acquisition Unit

If you are using an infrastructure Data Collector, you also need an Acquisition Unit to inject data into Cloud Insights. An Acquisition Unit is a computer dedicated to hosting data collectors, typically a Virtual Machine. This computer is typically located in the same data center/VPC as the monitored items.

### Telegraf Agents

Cloud Insights also supports Telegraf as its agent for collection of integration data. Telegraf is a plugin-driven server agent that can be used to collect and report metrics, events, and logs.

Cloud Insights Architecture



## Integration with Cloud Insights for VMs in Red Hat OpenShift Virtualization

To start collecting data for VMs in OpenShift Virtualization you will need to install:

1. A Kubernetes monitoring operator and data collector to collect Kubernetes data  
For complete instructions, refer to the [documentation](#).
2. An acquisition unit to collect data from ONTAP storage that provides persistent storage for the VM disks  
For complete instructions, refer to the [documentation](#).
3. A data collector for ONTAP  
For complete instructions, refer to the [documentation](#)

Additionally, if you are using StorageGrid for VM backups, you need a data collector for the StorageGRID as well.

## Sample Monitoring capabilities for VMs in Red Hat OpenShift Virtualization

### Monitoring based on events and creating Alerts

Here is a sample where the namespace that contains a VM in OpenShift Virtualization is monitored based on events. In this example, a monitor is created based on **logs.kubernetes.event** for the specified namespace in the cluster.

Observability

Explore

Alerts

Collectors

Log Queries

Enrich

Reporting

Kubernetes

Workload Security

ONTAP Essentials

Admin

NetApp PCS Sandbox / Observability / Alerts / Manage Monitors / Monitor virtual-machines-demo-ns

Edit log monitor

Filter/Advanced Query and Group by in section 1 must not be empty. If alert resolution is based on log entry, section 3 filter/advanced query also must not be empty.

Select the log to monitor

Log Source

logs.kubernetes.event

Filter By

kubernetes\_cluster

ocp-cluster4

involvedobject.namespace

virtual-machines-demo

Advanced Query

Group By

reason

27 items found

| timestamp              | type                  | source   | message                         |
|------------------------|-----------------------|--|---------------------------------|
| 04/19/2024 10:31:18 AM | logs.kubernetes.event | kubernetes_cluster:ocp-cluster4;namespace:cloudi<br>nsights-<br>monitoring;pod_name:net<br>app-ci-event-exporter-<br>7f7c8d84c4-sk7t9; | VirtualMachineInstance started. |
| 04/19/2024 10:31:18 AM | logs.kubernetes.event | kubernetes_cluster:ocp-cluster4;namespace:cloudi<br>nsights-<br>monitoring;pod_name:net<br>app-ci-event-exporter-<br>7f7c8d84c4-sk7t9; | VirtualMachineInstance defined. |

Define alert behavior

Create an alert at severity

Warning

when the conditions above occur

1

time

## Edit log monitor

### 1 Select the log to monitor

Log Source logs.kubernetes.event

7 items found

| timestamp ↓            | type                  | source   | message                         |
|------------------------|-----------------------|--|---------------------------------|
| 04/19/2024 10:31:18 AM | logs.kubernetes.event | kubernetes_cluster:ocp-cluster4;namespace:cloudi<br>nsights-monitoring;pod_name:net<br>app-ci-event-exporter-<br>7f7c8d84c4-sk7t9; | VirtualMachineInstance started. |
| 04/19/2024 10:31:18 AM | logs.kubernetes.event | kubernetes_cluster:ocp-cluster4;namespace:cloudi<br>nsights-monitoring;pod_name:net<br>app-ci-event-exporter-<br>7f7c8d84c4-sk7t9; | VirtualMachineInstance defined. |
| 04/19/2024 10:31:18 AM | logs.kubernetes.event | kubernetes_cluster:ocp-cluster4;namespace:cloudi<br>nsights-monitoring;pod_name:net<br>app-ci-event-exporter-<br>7f7c8d84c4-sk7t9; | VirtualMachineInstance started. |

## 2 Define alert behavior

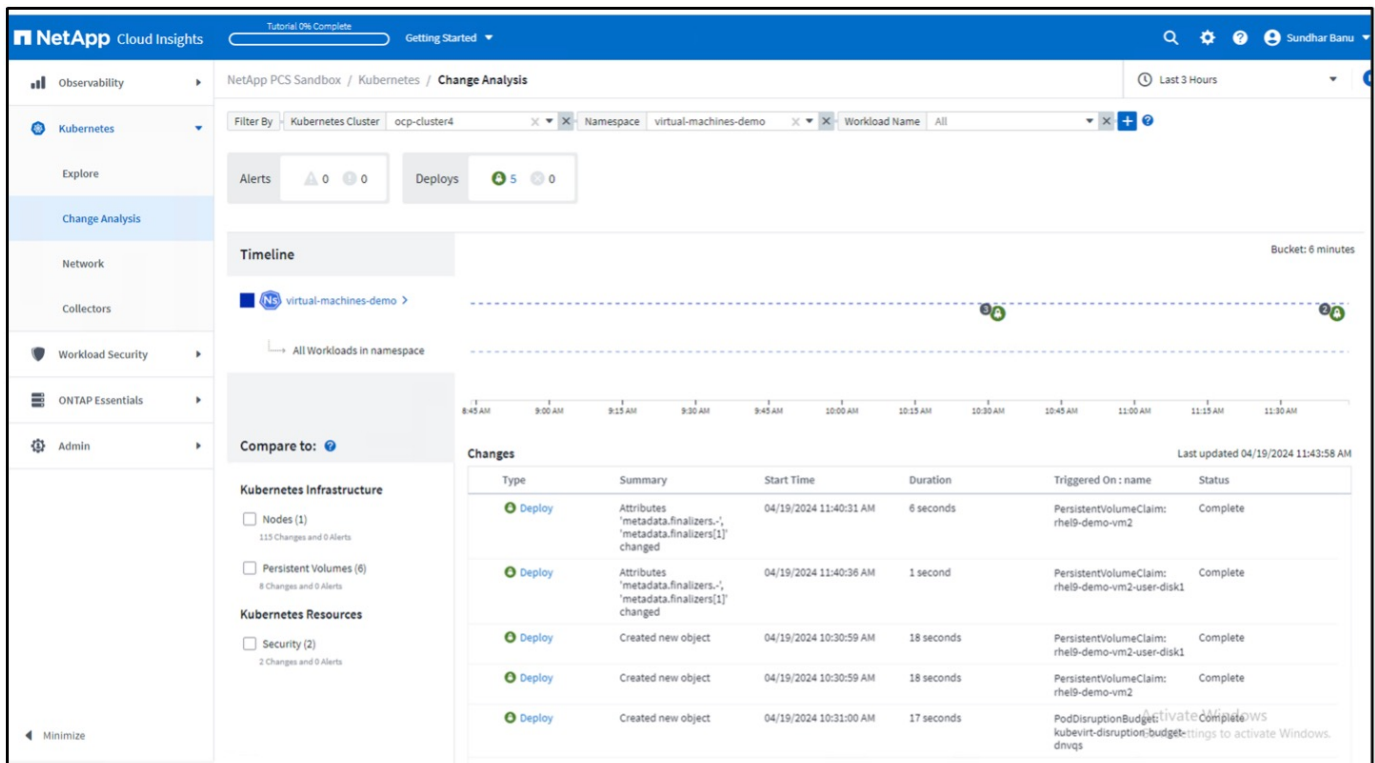
|                             |           |                                 |   |      |
|-----------------------------|-----------|---------------------------------|---|------|
| Create an alert at severity | Warning ▼ | when the conditions above occur | 1 | time |
|-----------------------------|-----------|---------------------------------|---|------|

This query provides all the events for the virtual machine in the namespace. (There is only one virtual machine in the namespace). An advanced query can also be constructed to filter based on the event where the reason is “failed” or “FailedMount” These events are typically created when there is an issue in creating a PV or mounting the PV to a pod indicating issues in the dynamic provisioner for creating persistent volumes for the VM.

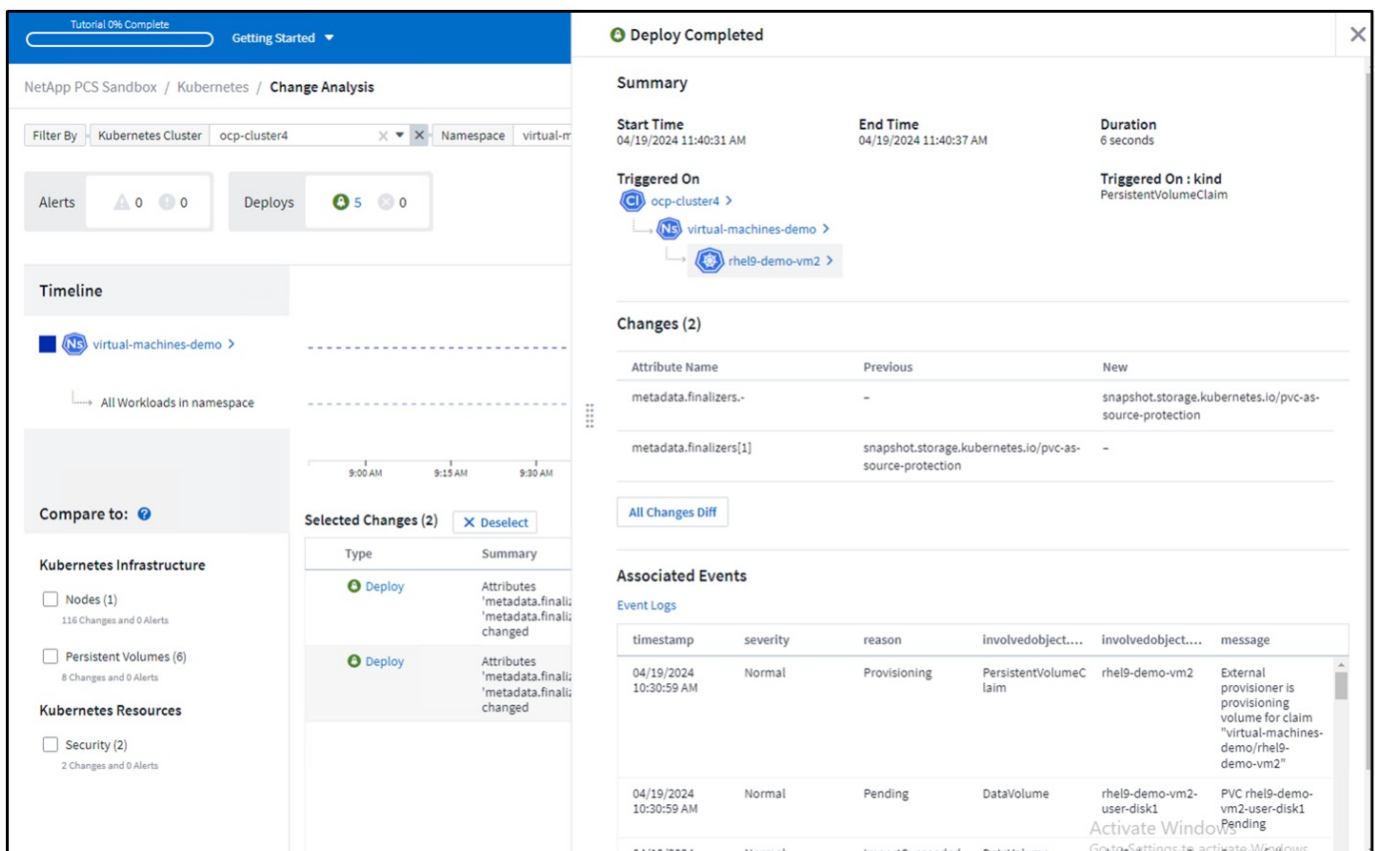
While creating the Alert Monitor as shown above, you can also configure notification to recipients. You can also provide corrective actions or additional information that can be useful to resolve the error. In the above example, additional information could be to look into the Trident backend configuration and storage class definitions for resolving the issue.

## Change Analytics

With Change Analytics, you can get a view of what changed in the state of your cluster including who made that change which can help in troubleshooting issues.



In the above example, Change Analysis is configured on the OpenShift cluster for the namespace that contains an OpenShift Virtualization VM. The dashboard shows changes against the timeline. You can drill down to see what changed and the click on All Changes Diff to see the diff of the manifests. From the manifest, you can see that a new backup of the persistent disks was created.



All Changes Diff

Previous

New

Expand 45 lines ...

46

kind: DataVolume

47

name: rhel9-demo-vm2

48

uid: dcf93b7a-71bc-409b-ad12-4916d05e0980

49

- resourceVersion: "8569671"

50

uid: 953a4188-5932-46ac-85d7-9734acc78278

51

spec:

52

accessModes:

Expand 15 lines ...

46

kind: DataVolume

47

name: rhel9-demo-vm2

48

uid: dcf93b7a-71bc-409b-ad12-4916d05e0980

49

+ resourceVersion: "8619670"

50

uid: 953a4188-5932-46ac-85d7-9734acc78278

51

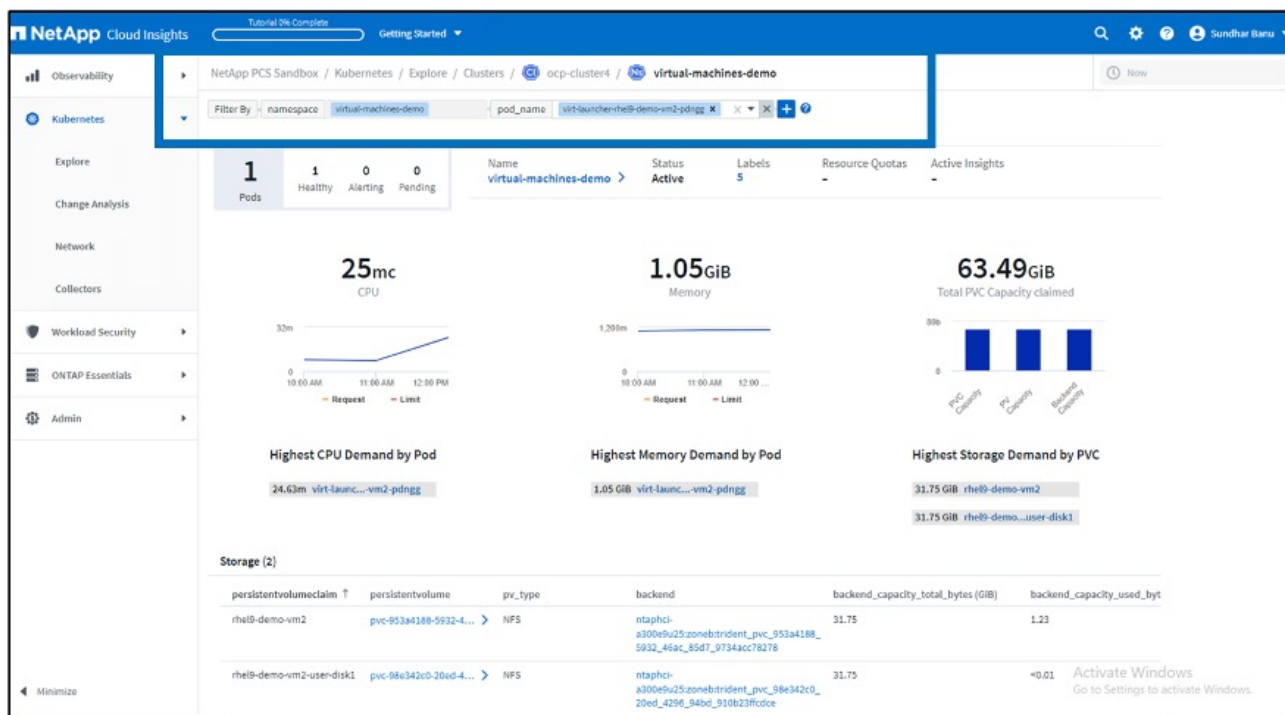
spec:

52

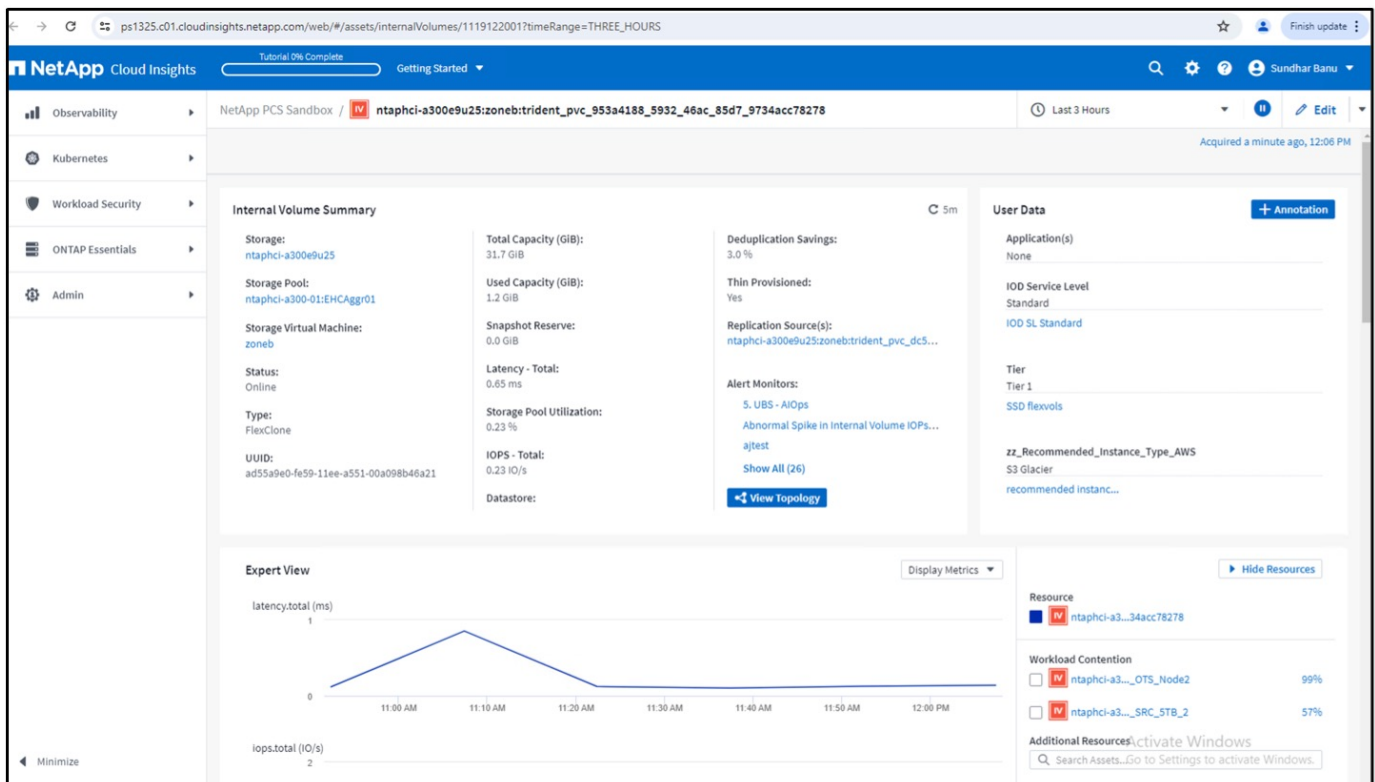
accessModes:

## Backend Storage Mapping

With Cloud Insights, you can easily see the backend storage of the VM disks and several statistics about the PVCs.



You can click on the links under the backend column, which will pull data directly from the backend ONTAP storage.



Another way to look at all the pod to storage mapping is creating an All Metrics query From Observability menu under Explore.

The screenshot shows the 'Explore' section of NetApp Cloud Insights. A query is configured for 'persistent disks' with filters for 'Object' (kubernetes.pod\_to\_storage), 'Filter by Attribute' (kubernetes\_cluster: op-cluster4), and 'Group By' (kubernetes.pod\_to\_storage). The results are displayed in a table with 6 items found.

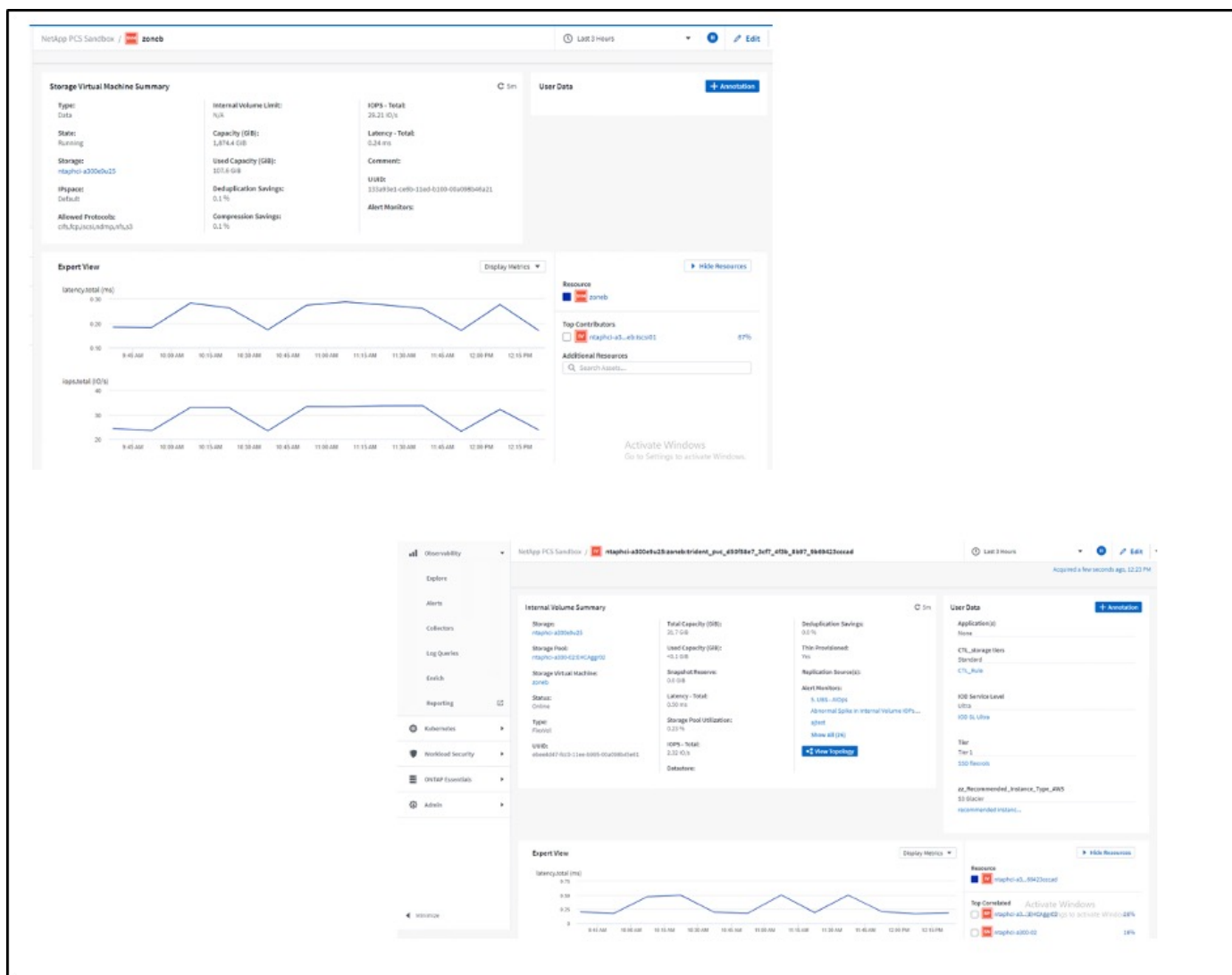
| Object                    | Filter by Attribute             | Group By                  |
|---------------------------|---------------------------------|---------------------------|
| kubernetes.pod_to_storage | kubernetes_cluster: op-cluster4 | kubernetes.pod_to_storage |

| Table Row Grouping                           | Metrics & Attributes  |
|--|---|
| kubernetes.pod_to_storage                    | persisten..., workload..., namespace, storagevirt..., InternalVol..., volume.na..., qtree.name, timeToFull..., backen |
| importer-prime-4f1b8351-2678-4295-b9db-64... | pvc-d4ccec2c-24b, openshift-virtualization-os-image, zoneb, ntaphci-a300e9u25, 3d72704c-6108-11e-000, 0.16            |
| importer-prime-8f792a30-02bb-4e86-a8a8-d6... | pvc-d50f58e7-3cf7, openshift-virtualization-os-image, zoneb, ntaphci-a300e9u25, 3d72704c-6108-11e-000, 0.16           |
| virt-launcher-rhel9-demo-vm2-pdngg           | pvc-98e342c0-20e, virtual-machines-demo, zoneb, ntaphci-a300e9u25, 3d72704c-6108-11e-000, 0.00                        |
| virt-launcher-rhel9-demo-vm2-pdngg           | pvc-953a4188-593, virtual-machines-demo, zoneb, ntaphci-a300e9u25, 3d72704c-6108-11e-000, 3.88                        |
| virt-launcher-rhel9-demo-vm2-rnztj           | pvc-f4d1ad3c-314, virtual-machines, zoneb, ntaphci-a300e9u25, 3d72704c-6108-11e-000, 3.88                             |
| virt-launcher-rhel9-demo-vm2-rnztj           | pvc-ad805a7b-4a1, virtual-machines, zoneb, ntaphci-a300e9u25, 3d72704c-6108-11e-000, 0.00                             |

Clicking on any of the links will give you the corresponding details from ONTP storage. For example, clicking on an SVM name in the storageVirtualMachine column will pull details about the SVM from ONTAP. Clicking on an internal volume name will pull details about the volume in ONTAP.



|                 |                         |                     |                                   |
|-----------------|-------------------------|---------------------|-----------------------------------|
|                 | storageVirtualMachin... | internalVolume.name | volume.na..                       |
| zation-os-image | zoneb                   |                     | ntaphci-a300e9u25:zoneb:trident_p |
| zation-os-image | zoneb                   |                     | ntaphci-a300e9u25:zoneb:trident_p |
| demo            | zoneb                   |                     | ntaphci-a300e9u25:zoneb:trident_p |
| demo            | zoneb                   |                     | ntaphci-a300e9u25:zoneb:trident_p |
|                 | zoneb                   |                     | ntaphci-a300e9u25:zoneb:trident_p |
|                 | zoneb                   |                     | ntaphci-a300e9u25:zoneb:trident_p |





## 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.