White Paper

Dell PowerProtect Data Manager: Protecting Kubernetes Workloads

Abstract

Kubernetes is an open-source platform to manage and orchestrate containerized workloads and services. This document describes the integration of Kubernetes workloads with Dell PowerProtect Data Manager and details how they protect Kubernetes workloads.

October 2022

Revisions

Date	Description
October 2020	Initial release
January 2021	Dell PowerProtect Data Manager 19.6 updates • PostgreSQL High Availability • Cassandra Application Consistent Protection • Restore Cluster Scoped Resources
February 2021	PowerProtect Data Manager 19.7 updates • VMware Tanzu Kubernetes Cluster protection
May 2021	PowerProtect Data Manager 19.8 Updates • Storage Class Mapping
May 2022	PowerProtect Data Manager 19.9 and 19.10 updates • Structural custom resource definitions (CRDs) • Quick Recovery to an alternate Kubernetes cluster
July 2022	PowerProtect Data Manager 19.11 updates • Integration with Dell ISG Storage • Protecting PVCs in PowerScale access zones • Support for encryption of backup and restore data in-flight for Kubernetes cluster assets
October 2022	PowerProtect Data Manager 19.12 updates • Add Cluster Root certificate in PowerProtect Data Manager UI • Update the PowerProtect Controller configuration, Velero configuration, or cProxy configuration fields in PowerProtect Data Manager UI • Download Kubernetes tools from PowerProtect Data Manager UI

Acknowledgments

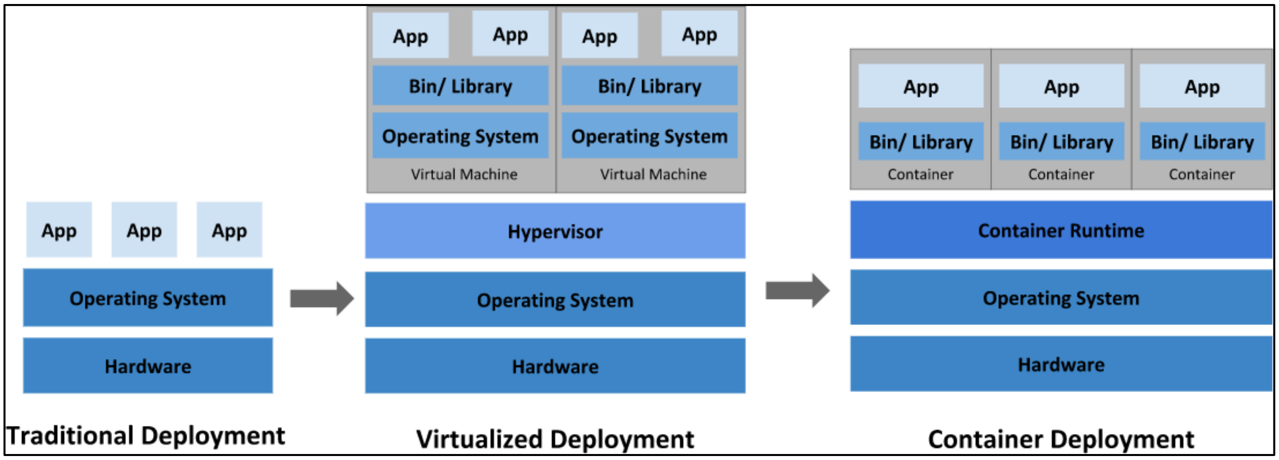
Author: Vinod Kumar Kumaresan

The information in this publication is provided “as is.” Dell Inc. makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose.

Table of contents

Executive summary

Traditionally, organizations used physical servers to run applications. There was no way to define resource boundaries for applications in a physical server, and this caused resource allocation issues. As a solution, virtualization was introduced, making it possible to run multiple virtual machines (VMs) on a single physical server's CPU. It also enabled applications to be isolated within VMs and provided a level of security.



Modern infrastructure is being transformed by containers. Containers are like virtual machines but have relaxed isolation properties to share the operating system. The container has its own file system, CPU, memory, and process space. Agile application creation, continuous development, environmental consistency across development, application-centric management, efficient resource allocation, and resource isolation are the key benefits of containers. Kubernetes is an open-source container management platform that unifies a cluster of machines into a single pool of compute resources.

With a distributed container deployment, it is important to protect your workloads. Dell PowerProtect Data Manager protects the Kubernetes workloads and ensures high availability, and consistent and reliable backup and restore capabilities for Kubernetes workloads or for a disaster recovery (DR) situation. PowerProtect Data Manager offers centralized management, automation, multi-cloud options, and advanced integration to simplify managing workloads.

Audience

This white paper is intended for customers, partners, and other users who want to understand how PowerProtect Data Manager helps protect Kubernetes workloads.

1 Introduction

Kubernetes is an open-source container orchestrator for managing containerized workloads and services, enabling both declarative configuration and automation. It is portable, extensible, and scalable and has a large, rapidly growing ecosystem. Kubernetes services, support, and tools are widely available.

Dell PowerProtect Data Manager protects existing and newly discovered workloads. It allows IT operations and backup administrators to manage Kubernetes clusters and their protection through a single management UI and define protection policies for Kubernetes workloads from Kubernetes APIs. The policy-driven protection is defined by the protection policy mechanism. PowerProtect Data Manager discovers the namespaces, labels, and pods in the environment, and helps protect them by providing cluster credentials logging, monitoring, governance, and recovery.

1.1 Features of Kubernetes

Kubernetes is an open-source system for automating deployment, scaling, and management of containerized applications.

- Kubernetes automates Linux container operations and eliminates many of the manual processes involved in deploying and scaling containerized applications.
- Applications can be clustered together in group of hosts running Linux containers, and Kubernetes helps you easily and efficiently manage those clusters.
- Kubernetes is an ideal platform for hosting cloud-native applications that require rapid scaling.

1.2 PowerProtect Data Manager capabilities for Kubernetes

PowerProtect Data Manager provides enterprise-level protection for Kubernetes.

1.2.1 Efficient and flexible

- PowerProtect Data Manager provides a single platform for data protection. It manages different workloads such as VMs, applications, and the containers through one platform.
- Protection to deduplicated storage allows great total cost of ownership (TCO) with PowerProtect DD series.

1.2.2 Built for Kubernetes

PowerProtect Data Manager enables the following:

- Provides flexible protection for Kubernetes clusters using the Kubernetes APIs
- Discovers, monitors, and protects Kubernetes resources—namespaces, labels, pods, and persistent volumes
- Does not require installing a backup client container for each pod for the backup process
- Provides protection to controllers per node to avoid cross-node traffic
- Enables application consistency for MySQL and MongoDB databases
- Can restore assets to another cluster that is connected to PowerProtect Data Manager
- Provides protection for AWS-hosted Kubernetes clusters using PowerProtect Data Manager running on AWS and protected to PowerProtect DDVE running on AWS
- Data in-flight encryption for Kubernetes

1.3 Key components of PowerProtect Data Manager

1.3.1 Cloud Native Data Manager (CNDM)

The Cloud Native Data Manager (CNDM) is an integrated microservice component of PowerProtect Data Manager which communicates with the kube-apiserver of the cluster. This component is responsible for APIs for the backup and restore process.

1.3.2 PowerProtect controller

PowerProtect controller is the component which is installed on the Kubernetes cluster when the cluster is discovered by PowerProtect Data Manager. The backup and restore controllers manage BackupJob CR and RestoreJob CR definitions. The PowerProtect controller is responsible for the backup and restore of persistent volumes.

1.3.3 VMware Velero

VMware Velero is the open-source tool which is integrated with PowerProtect Data Manager. It is integrated and is not required to be installed separately. The Velero component is pushed into the Kubernetes cluster by the PowerProtect controller pod after the cluster is in the running state using the Velero deployment object. It is responsible for the backup and restore of metadata.

1.3.4 cProxy

The containerized proxy (cProxy) is a stateless containerized proxy. It is installed on the Kubernetes cluster when the backup and restore process is initiated and is deleted after the process is completed. It is responsible for managing persistent volume snapshots (snap copies), mounting snapshots, and moving the data to the target storage. It is also responsible for restoring data into a persistent volume from target storage and making the data available for attaching to pods.

1.4 Key components of Kubernetes

1.4.1 Cluster

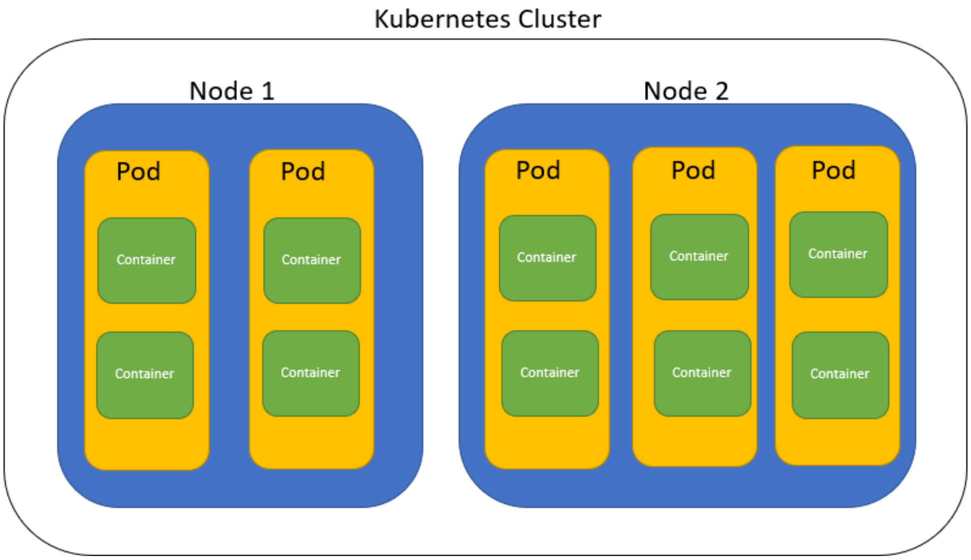
A Kubernetes cluster is a group of machines called nodes that run containerized applications and has a wanted state that defines which applications or workloads should be running. The cluster’s wanted state is defined with the Kubernetes API.

1.4.2 Node

A node is defined for a virtual or physical machine, depending on the cluster. Each node contains the services necessary to run pods and is managed by the control-plane components. There are two kinds of nodes: control-plane node and worker node.

1.4.3 Pods and containers

Pods operate at one level above the individual container. Multiple containers can be encapsulated within a pod. A Kubernetes pod is a group of containers that are deployed together on the same host. The pod is sometimes called as container when a single container is frequently deployed.



1.4.4 Kubernetes API (kube-apiserver)

The Kubernetes API server is the control-plane of the Kubernetes cluster that exposes the Kubernetes API. It serves as the foundation for the declarative configuration schema for the system. The **kubectl** command-line tool can be used to create, update, delete, and get API objects.

1.4.5 Persistent volume and persistent volume claim

A persistent volume (PV) is storage defined for the cluster that is provisioned by an administrator or dynamically provisioned using storage classes (SCs). It is a resource in the cluster similar to a node. PVs are volume plug-ins like volumes but have a life cycle independent of any individual pod that uses the PV. It captures the details of the implementation of the storage that is NFS, iSCSI, or a cloud-provider-specific storage system.

A persistent volume claim (PVC) is a request for storage by a user. It is like a pod, which consumes node resources. Similarly, PVCs consume PV resources. Pods can request specific levels of resources (CPU and memory).

1.4.6 Container storage interface

Container storage interface (CSI) defines a standard interface for container orchestration systems to expose arbitrary storage systems to respective container workloads. A CSI-compatible volume driver is deployed on a Kubernetes cluster so that users can use the CSI volume type to attach or mount the volumes exposed by the CSI driver.

1.4.7 Storage class

A storage class is described as the type of storage that is provisioned and allowed ranges for size and IOPS. When user creates a PVC, that specifies the storage class with size in GB and number of IOPS. A storage class is used to abstract the underlying storage platform.

1.4.8 Namespaces

A namespace is defined as Kubernetes object which partitions a single Kubernetes cluster into multiple virtual clusters. Namespaces are intended for the use in environment with many users spread across multiple teams or projects.

1.4.9 Custom resource

A resource in the Kubernetes environment is an endpoint for an API that stores a collection of API objects of a certain kind. A custom resource (CR) is an extension of the Kubernetes API that is not necessarily available in a default Kubernetes installation. It represents a customization of a particular Kubernetes installation.

2 Deployment methods of Kubernetes

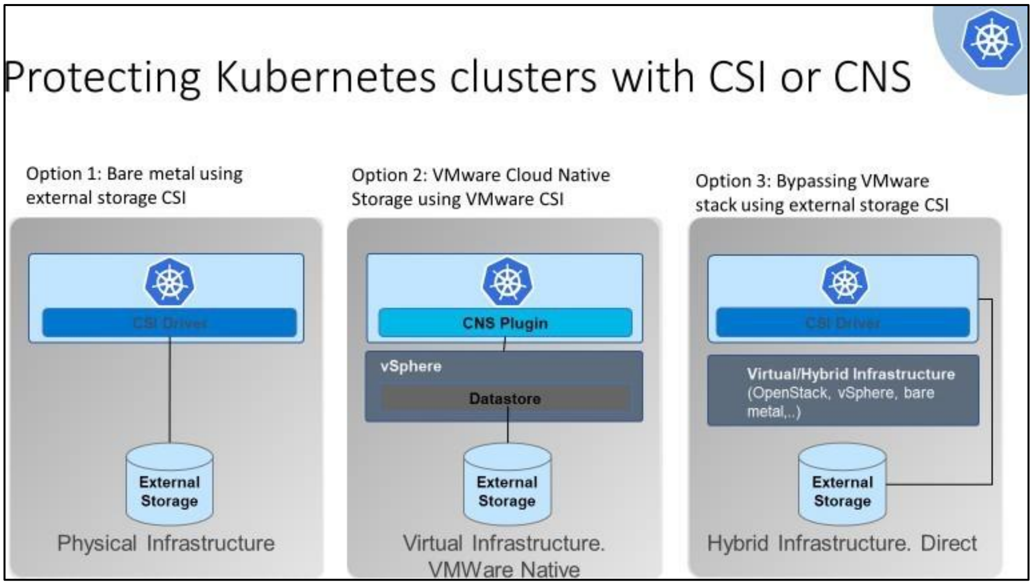
Kubernetes is an open-source container orchestrator for automating deployment, scaling, and managing containerized workloads. There are few methods to deploy Kubernetes clusters and protected accordingly. Kubernetes can be deployed on-premises or on the cloud.

2.1 Kubernetes on-premises

Kubernetes is described as a cloud-native technology. However, the cloud-native concept includes the use of on-premises infrastructure.

2.1.1 Kubernetes running on virtual environment

The virtual environments are mostly VMware based. With this method of deployment, the persistent volumes are carved with VMware vSphere managed storage—VMware vSAN or VMware Virtual Volumes (vVols)—as a first-class disk. A first-class disk (FCD) refers to an improved virtual disk (IVD) which is a feature of VMware vSphere. The FCD is the independent disk which is not associated with VM. When Persistent Volumes for cloud native application is created, a virtual disk (VMDK) is attached to Kubernetes node. The disk backup of FCD is similar to that of other VMDKs.



Protecting Kubernetes clusters with CSI or CNS

Option 1: Bare metal using external storage CSI

Option 2: VMware Cloud Native Storage using VMware CSI

Option 3: Bypassing VMware stack using external storage CSI

CSI Driver
External Storage
Physical Infrastructure
CNS Plugin
vSphere
Datastore
External Storage
Virtual Infrastructure. VMWare Native
CSI Driver
Virtual/Hybrid Infrastructure (OpenStack, vSphere, bare metal,..)
External Storage
Hybrid Infrastructure. Direct

The container orchestrates volume snapshot backup with which the volume snapshot is taken, mounted, and streamed block data to target storage. For application-level backups, pre- and post-hooks are used which quiesce the database, flush, and take snapshots of Persistent Volumes.

2.1.2 On-premises Kubernetes on bare metal

Bare-metal servers, such as Linux servers, run containers and Kubernetes nodes on physical servers. In this method of deployment, the Persistent Volumes are carved out from underlying storage that is SAN or NAS and connected through the storage vendor provided custom drivers or through standardized CSI drivers. The container orchestrates volume snapshot backup with which the volume snapshot is taken, mounted, and streamed block data to target storage. For application-level backups, the database dump backup is taken online and streamed data to target storage.

2.1.3 Using external CSI

Container storage interface (CSI) uses an open standard API that enables Kubernetes to expose arbitrary storage systems to containerized workloads. In this method of deployment, external CSI drivers are used to access infrastructure and external storage such as OpenStack.

2.2 Kubernetes on Cloud

There are few clouds which are supported for Kubernetes. Database backups are taken online and streamed data to cloud bucket. The container orchestrates cloud snapshots. Kubernetes deployment in the cloud that is Amazon Web Services (AWS), Microsoft Azure or Google Cloud have two variants.

2.2.1 Kubernetes deployed on Infrastructure as a Service (IaaS)

In this scenario, the Linux hosts running containers are deployed on IaaS plane (for example, AWS EC2 instance), which means the underlying infrastructure is provided and is similar to running on a virtualized environment. In such case, the user must deploy the Kubernetes clusters either upstream or through a distribution on the IaaS instance of the cloud provider.

2.2.2 Kubernetes as a Service (KaaS)

Various cloud providers offer Kubernetes as a service. In such case, the cloud provider manages the Kubernetes environment completely and offers it as a service to the user. Below is the list of cloud providers and the Kubernetes as a service:

- Kubernetes on Google Cloud: Google Kubernetes Engine (GKE)
- Kubernetes on Amazon Web Services (AWS): Elastic Kubernetes Service (EKS)
- Kubernetes on Microsoft: Azure Kubernetes Service (AKS)
- Kubernetes on Rancher: Rancher Kubernetes Engine (RKE)

Also, the Kubernetes environment can be a generic upstream instance that is an open-source version of Kubernetes, Distro (Kubernetes distribution) with Rancher and CoreOS. It automates the provisioning of the Kubernetes cluster using an installer script and OEM distribution that automates Kubernetes, which is Red Hat OpenShift, Anthos, or Pivotal Container Service (PKS).

3 Reference architecture

PowerProtect Data Manager protects Kubernetes workloads and ensures the data is consistent and highly available. PowerProtect Data Manager is a virtual appliance that is deployed on an ESXi host using OVA and is integrated with PowerProtect DD series appliances as protection target where backups are stored.

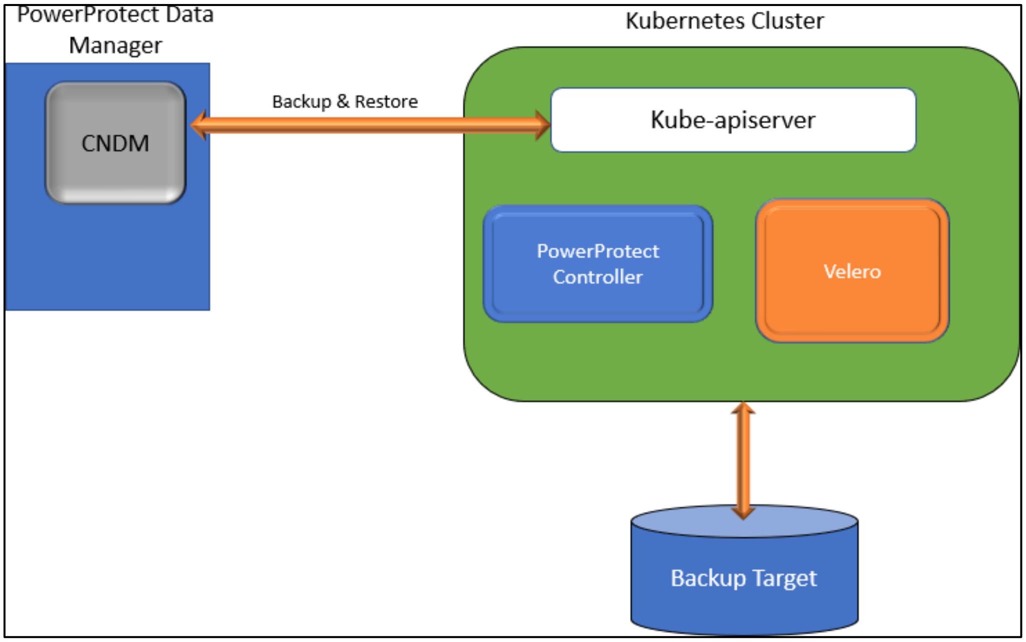
A Kubernetes cluster is group of a control-plane and worker nodes. The control-plane node manages the worker nodes, and it handles scheduling the pods across the nodes in the cluster. PowerProtect Data Manager integrates with Kubernetes cluster through Kubernetes APIs to perform the discovery. The CNDM is the component of PowerProtect Data Manager which communicates with the kube-apiserver of the cluster.

When the cluster is discovered, the cluster is added as PowerProtect Data Manager asset source and associated namespaces as assets are available to be protected. During the process of the discovery, PowerProtect Data Manager creates the following two namespaces in the cluster. The data is compressed and deduplicated at the source and sent to the target storage.

- **Velero-ppdm**: Contains a Velero pod to backup metadata and stage to the target storage if there is a BareMetal environment. It performs PVC and metadata backup if there is VMware Cloud Native Storage (CNS).
- **PowerProtect**: Contains a PowerProtect controller pod to drive Persistent Volume Claim snapshot and backup and push the backups to target storage using intermittently spawned cProxy pods.

3.1 Protection of Kubernetes clusters

PowerProtect Data Manager discovers the Kubernetes clusters using the IP address or FQDN. PowerProtect Data Manager uses the discovery service account and the token (kubeconfig file) to integrate with kube-apiserver. PowerProtect Data Manager protects two types of assets of Kubernetes cluster that are Namespaces and PersistentVolumeClaims (PVCs). The PowerProtect controller is responsible for the backup and restore of PVCs. The VMware Velero component is responsible for the backup and restore of metadata that is saved in the target storage.



A protection policy enables selecting a specific group of assets to be backed up. The protection policy is created using the PowerProtect Data Manager UI for the backup schedule and retention lock. When a protection policy is created, a new storage unit (SU) is created on the protection storage as part of protection policy configuration.

With Kubernetes workloads, a BackupStorageLocation containing the SU information is also created on the cluster. The PowerProtect controller running in the Kubernetes cluster creates a corresponding BackupStorageLocation in the Velero namespace whenever a BackupStorageLocation is created in the PowerProtect namespace.

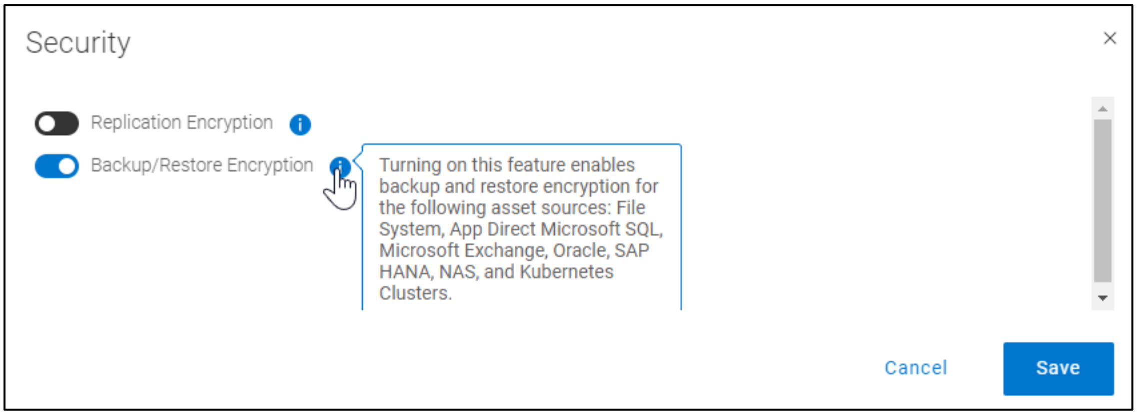
3.2 Structured custom resource definitions

Starting with version 19.9, PowerProtect Data Manager uses structured customer resource definitions (CRDs) with Kubernetes to increase system security. PowerProtect Data Manager versions earlier than 19.9 used non-structural schemas that did not validate the consistent format of custom resources (CRs). When updating from older versions of PowerProtect Data Manager, the self-service recovery of the last backup copy taken before the update is not possible. To restore that backup, use the PowerProtect Data Manager user interface. There are important Kubernetes-specific factors to consider when updating PowerProtect Data Manager in a Kubernetes environment. For more information, see the Dell PowerProtect Data Manager Kubernetes User Guide.

Note: For general information about updating PowerProtect Data Manager, see the Dell PowerProtect Data Manager Kubernetes User Guide.

3.3 Data in-flight encryption for Kubernetes

Encryption of backup and restore data in-flight is now available for Kubernetes cluster assets. This functionality can be enabled in the PowerProtect Data Manager UI **System Settings > Security** dialog box.



Security

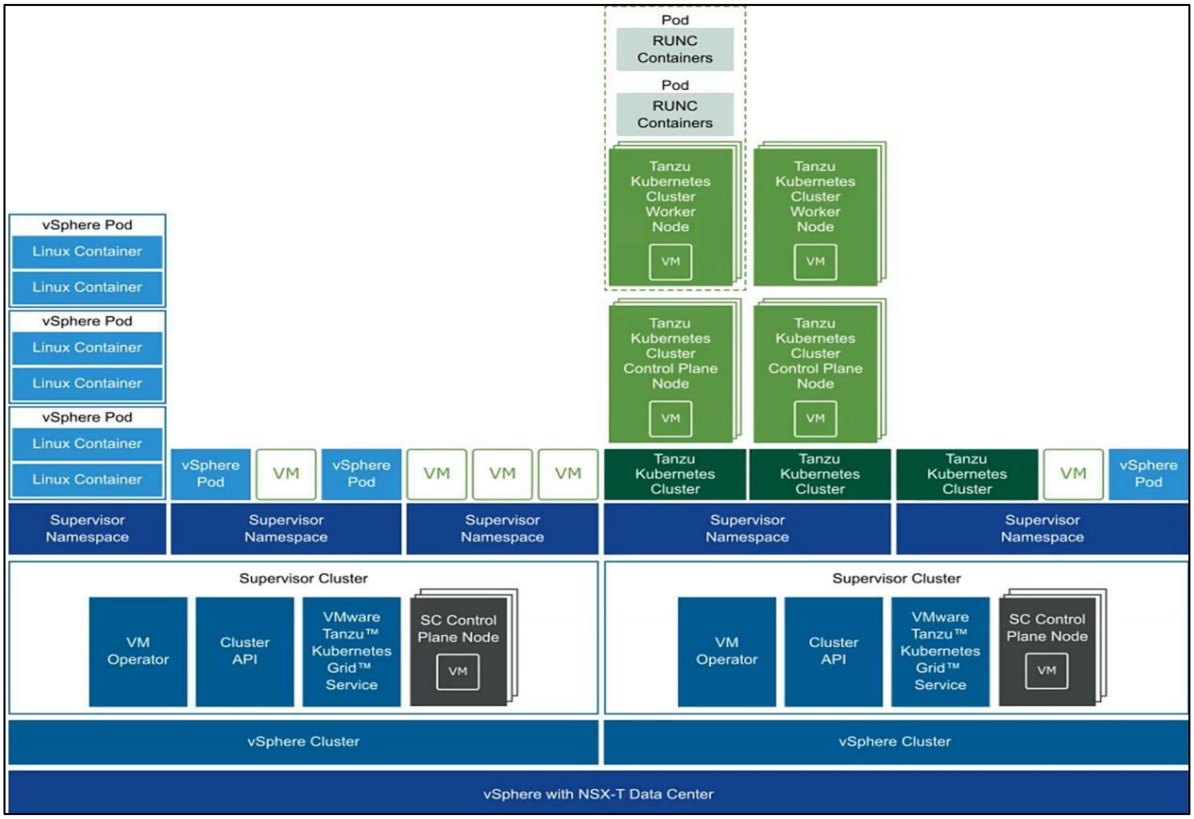
○ Replication Encryption

● Backup/Restore Encryption

Turning on this feature enables backup and restore encryption for the following asset sources: File System, App Direct Microsoft SQL, Microsoft Exchange, Oracle, SAP HANA, NAS, and Kubernetes Clusters.

3.4 Protection of VMware Tanzu Kubernetes Grid

Data Manager 19.7 onwards introduces ability to protect Kubernetes workloads on VMware Tanzu Kubernetes Grid (TKG). VMware vSphere 7U1 re-architectures vSphere with native Kubernetes as its control plane. A TKG cluster is a Kubernetes cluster that runs inside the virtual machines on supervisor layer which allows to run Kubernetes with consistency. It is enabled through the TKG service for VMware vSphere and is upstream-complaint with open-source Kubernetes (guest cluster). The guest cluster is a consistent Kubernetes cluster that runs on VMs and consists of its control-plane VM, management plane VM, worker nodes, pods, and containers.



Note: For more information about Dell PowerProtect Data Manager protecting VMware Tanzu Kubernetes Clusters, see the document PowerProtect Data Manager: Protecting VMware Tanzu Kubernetes Clusters.

4 Configuring PowerProtect Data Manager for protecting Kubernetes workloads

4.1 Adding a Kubernetes cluster asset source

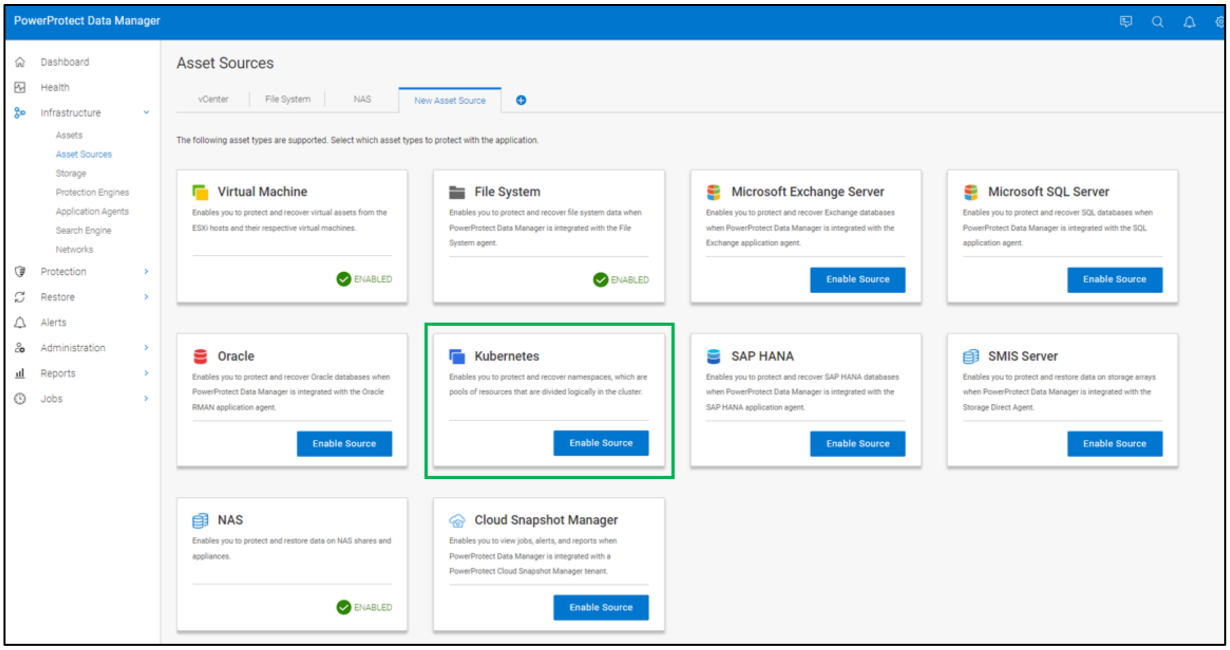
Adding a Kubernetes cluster as an asset source in PowerProtect Data Manager enables you to protect namespaces and persistent volume claims (PVCs) within the cluster. You can use the asset sources window in the PowerProtect Data Manager UI to add a Kubernetes cluster asset source.

See the section “Prerequisites to Kubernetes cluster discovery” in the PowerProtect Data Manager Kubernetes User Guide before adding a Kubernetes cluster as an asset source with Data Manager.

4.1.1 Enable Kubernetes asset source

An asset source must be enabled in PowerProtect Data Manager before you can add and register the asset source for the protection of assets.

The Kubernetes asset source can be enabled from the PowerProtect Data Manager UI. Click **Infrastructure** > **Asset Sources**, and click **+ (plus)** to view the **New Asset Source** tab. In the pane for the asset source that you want to add, click **Enable Source**. The **Asset Sources** window updates to display a tab for the new asset source.



The following asset types are supported. Select which asset types to protect with the application.

Virtual Machine
Enables you to protect and recover virtual assets from the ESXi hosts and their respective virtual machines.
✓ ENABLED

File System
Enables you to protect and recover file system data when PowerProtect Data Manager is integrated with the File System agent.
✓ ENABLED

Microsoft Exchange Server
Enables you to protect and recover Exchange databases when PowerProtect Data Manager is integrated with the Exchange application agent.
Enable Source

Microsoft SQL Server
Enables you to protect and recover SQL databases when PowerProtect Data Manager is integrated with the SQL application agent.
Enable Source

Oracle
Enables you to protect and recover Oracle databases when PowerProtect Data Manager is integrated with the Oracle RMAN application agent.
Enable Source

Kubernetes
Enables you to protect and recover namespaces, which are pools of resources that are divided logically in the cluster.
Enable Source

SAP HANA
Enables you to protect and recover SAP HANA databases when PowerProtect Data Manager is integrated with the SAP HANA application agent.
Enable Source

SMIS Server
Enables you to protect and restore data on storage arrays when PowerProtect Data Manager is integrated with the Storage Direct Agent.
Enable Source

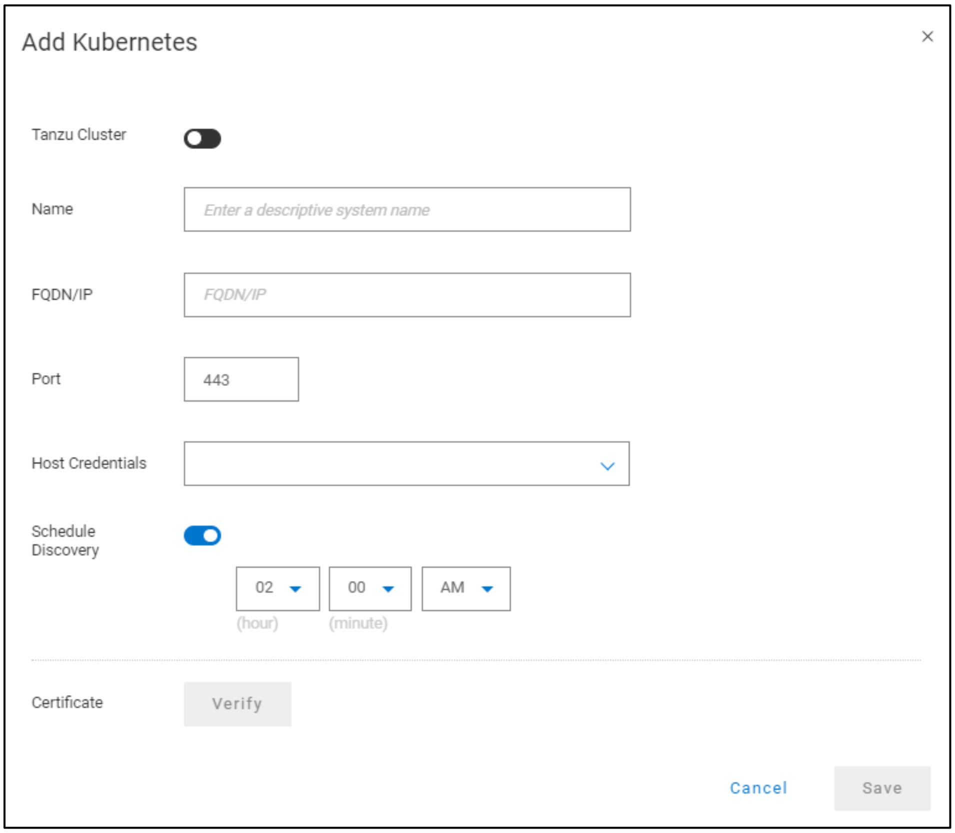
NAS
Enables you to protect and restore data on NAS shares and appliances.
✓ ENABLED

Cloud Snapshot Manager
Enables you to view jobs, alerts, and reports when PowerProtect Data Manager is integrated with a PowerProtect Cloud Snapshot Manager tenant.
Enable Source

4.1.2 Adding a Kubernetes cluster

A Kubernetes cluster can be added as an asset source in PowerProtect Data Manager to protect the namespaces and PVCs within the cluster.

Note: Discovery of a Kubernetes cluster discovers namespaces that contain volumes from both container storage interface (CSI) and non-CSI based storage. However, backup and recovery are supported only from CSI-based storage. Also, only PVCs with the VolumeMode Filesystem are supported.



Add Kubernetes ×

Tanzu Cluster

Name: *Enter a descriptive system name*

FQDN/IP: *FQDN/IP*

Port: 443

Host Credentials

Schedule Discovery

02 (hour) 00 (minute) AM

Certificate: Verify

Cancel Save

Name: Cluster name

Address: The fully qualified domain name (FQDN) or the IP address of the Kubernetes API server.

Note: We recommend using the FQDN instead of the IP address.

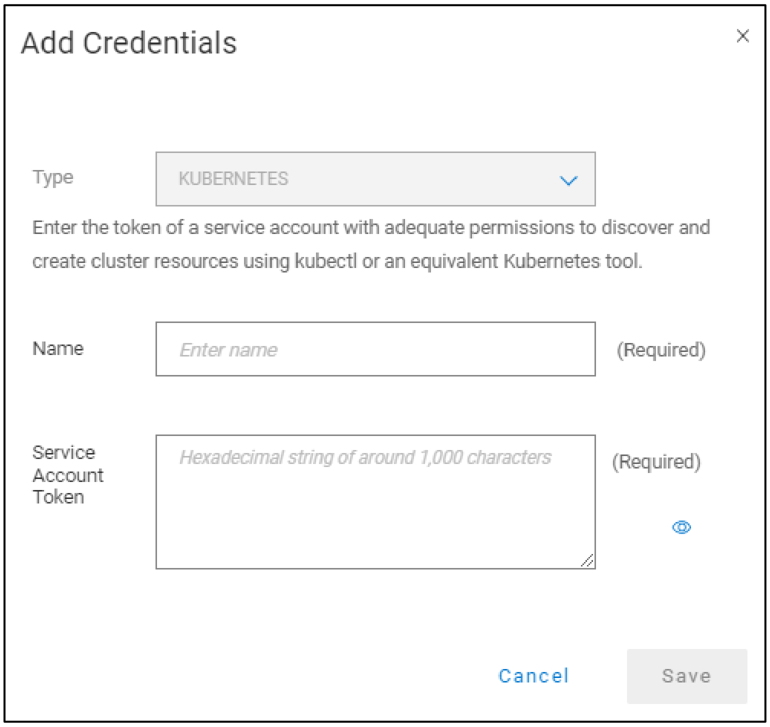
Port: Specify the port to use for communication when not using the default port, 443.

Note: The use of any port other than 443 or 6443 requires you to open the port on PowerProtect Data Manager first to enable outgoing communication.

Host Credentials: The service account must have the following privileges:

- Get/Create/Update/List CustomResourceDefinitions
- Get/Create/Update ClusterRoleBinding for 'cluster-admin' role

- Create/Update the 'powerprotect' namespace
- Get/List/Create/Update/Delete/List
- Get/List/Create/Update/Delete all kinds of resources inside the 'powerprotect' namespace
- Get/List/Watch all namespaces in the cluster and PV, PVC, storageclass, deployments, and pods in these namespaces



Add Credentials

Type: KUBERNETES

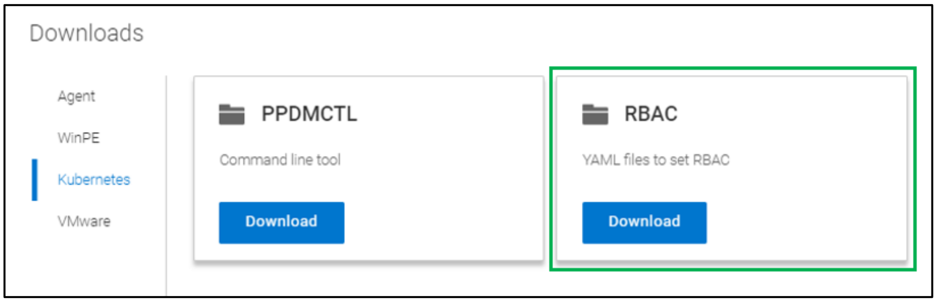
Enter the token of a service account with adequate permissions to discover and create cluster resources using kubectl or an equivalent Kubernetes tool.

Name: Enter name (Required)

Service Account Token: Hexadecimal string of around 1,000 characters (Required)

Note: The admin-user service account in the kube-system namespace contains all these privileges. You can provide the token of this account, or an existing similar service account. Alternatively, create a service account that is bound to a cluster role that contains these privileges, and then provide the token of this service account.

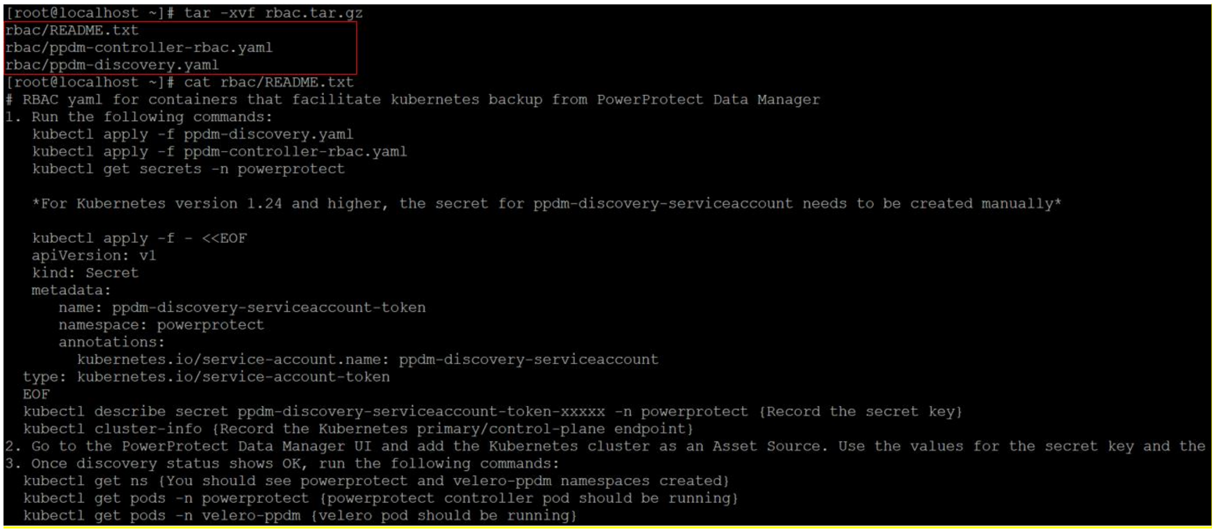
If you do not want to provide a service account with cluster-admin privileges, download the yaml files from the PowerProtect Data Manager UI **Downloads** window by clicking the **System Settings** icon and selecting **Downloads**. These files provide the definition of the cluster role with the required privileges required for PowerProtect Data Manager. Follow the instructions in the `README.txt` within the tar file to create the required clusterroles and clusterrolebindings, and to provide the token of the service account created in the yaml files. The README.txt file also provides instructions for manually creating the secret for `ppdm-discovery-serviceaccount`, which is required in Kubernetes versions 1.24 and later.



Downloads

PPDMCTL: Command line tool

RBAC: YAML files to set RBAC



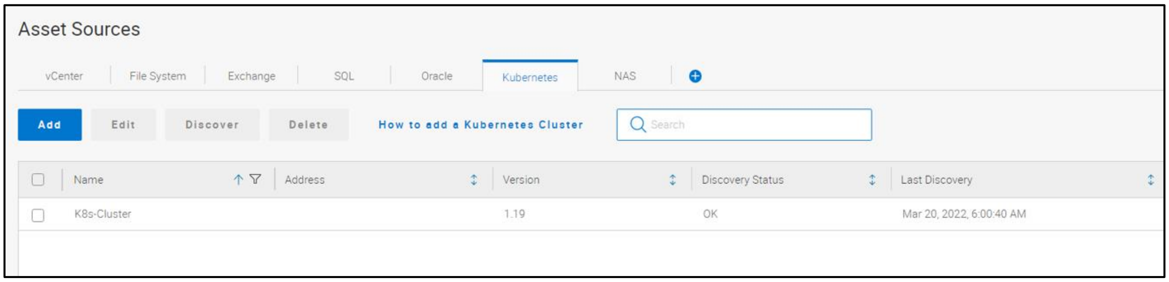
```
[root@localhost ~]# tar -xvf rbac.tar.gz
rbac/README.txt
rbac/ppdm-controller-rbac.yaml
rbac/ppdm-discovery.yaml
[root@localhost ~]# cat rbac/README.txt
# RBAC yaml for containers that facilitate kubernetes backup from PowerProtect Data Manager
1. Run the following commands:
   kubectl apply -f ppdm-discovery.yaml
   kubectl apply -f ppdm-controller-rbac.yaml
   kubectl get secrets -n powerprotect

   *For Kubernetes version 1.24 and higher, the secret for ppdm-discovery-serviceaccount needs to be created manually*

   kubectl apply -f - <<EOF
   apiVersion: v1
   kind: Secret
   metadata:
      name: ppdm-discovery-serviceaccount-token
      namespace: powerprotect
      annotations:
        kubernetes.io/service-account.name: ppdm-discovery-serviceaccount
  type: kubernetes.io/service-account-token
  EOF
  kubectl describe secret ppdm-discovery-serviceaccount-token-xxxxx -n powerprotect {Record the secret key}
  kubectl cluster-info {Record the Kubernetes primary/control-plane endpoint}
2. Go to the PowerProtect Data Manager UI and add the Kubernetes cluster as an Asset Source. Use the values for the secret key and the
3. Once discovery status shows OK, run the following commands:
  kubectl get ns {You should see powerprotect and velero-ppdm namespaces created}
  kubectl get pods -n powerprotect {powerprotect controller pod should be running}
  kubectl get pods -n velero-ppdm {velero pod should be running}
```

When adding the Kubernetes cluster as an asset source, a PowerProtect controller is installed on the cluster. This controller is also used to install Velero with the DD object-store plug-in and the vSphere plug-in.

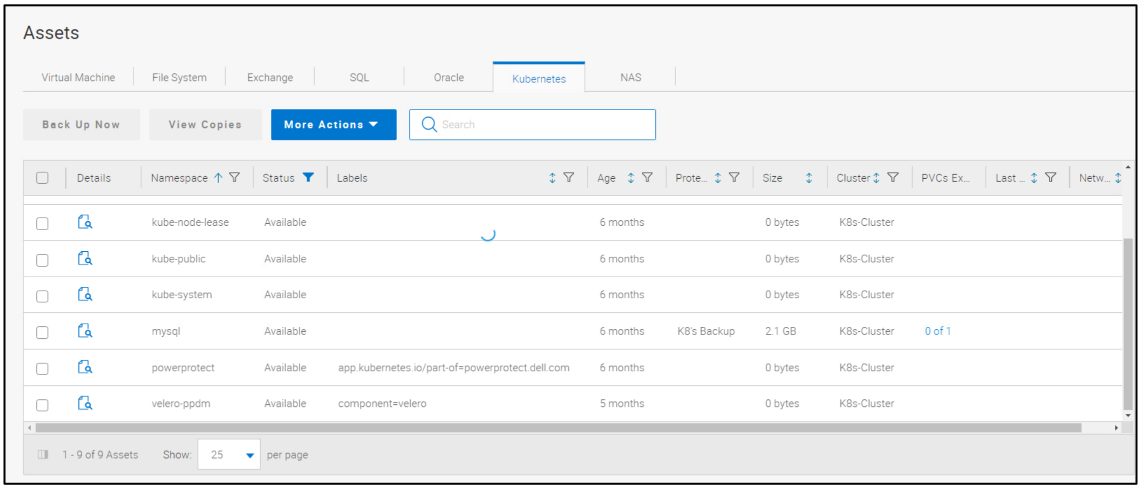
For more details, see the section “Add a Kubernetes cluster” in the PowerProtect Data Manager Kubernetes User Guide.



Asset Sources

□	Name	Address	Version	Discovery Status	Last Discovery
□	K8s-Cluster		1.19	OK	Mar 20, 2022, 6:00:40 AM

The namespaces in the Kubernetes cluster appear in the Kubernetes tab of the **Assets** window.



Assets

□	Details	Namespace	Status	Labels	Age	Prote...	Size	Cluster	PVCs Ex...	Last ...	Netw...
□		kube-node-lease	Available		6 months		0 bytes	K8s-Cluster			
□		kube-public	Available		6 months		0 bytes	K8s-Cluster			
□		kube-system	Available		6 months		0 bytes	K8s-Cluster			
□		mysql	Available		6 months	K8's Backup	2.1 GB	K8s-Cluster	0 of 1		
□		powerprotect	Available	app.kubernetes.io/part-of=powerprotect.dell.com	6 months		0 bytes	K8s-Cluster			
□		velero-ppdm	Available	component=velero	5 months		0 bytes	K8s-Cluster			

1 - 9 of 9 Assets Show: 25 per page

4.1.3 Adding cluster Root certificate in PowerProtect Data Manager UI

If the Kubernetes cluster is set up in high availability mode and the Kubernetes API server is not configured to send the root certificate as part of the TLS communication setup, backup and restore operations might fail. To resolve this issue, the Kubernetes cluster root certificate needs to be added to the PowerProtect Data Manager server. Starting with PowerProtect Data Manager version 19.12, the root certificate can be added in the PowerProtect Data Manager UI as shown here.

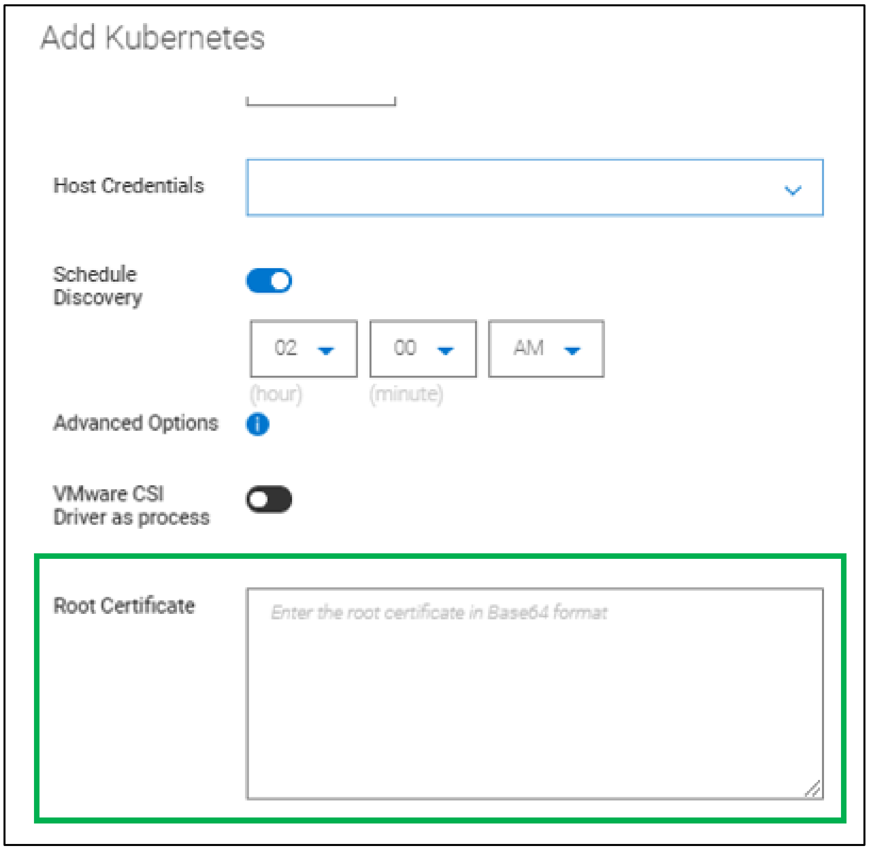
1. Copy the root certificate of the Kubernetes cluster to the PowerProtect Data Manager server.

2. In the PowerProtect Data Manager UI, go to **Infastructure > Asset Sources**.

3. Under the Kubernetes tab, select the Kubernetes cluster asset source and click **Edit**.

4. Expand **Advanced Options**, and then copy the text of the root certificate (in Base64 format) into the Root Certificate box.

5. Click **Save**.



The root certificate can be obtained by running the following command:

On AWS EKS, run `aws eks describe-cluster --region region --name Kubernetes cluster name --query "cluster.certificateAuthority.data" --output certificate file name`

For other distributions, run `kubectl config view --flatten` or its equivalent and obtain the Base64 encoded root certificate from the `certificate-authority-data` field for the cluster.

Note: This step is only required for other distributions when certificate-related errors occur while adding the Kubernetes cluster asset source.

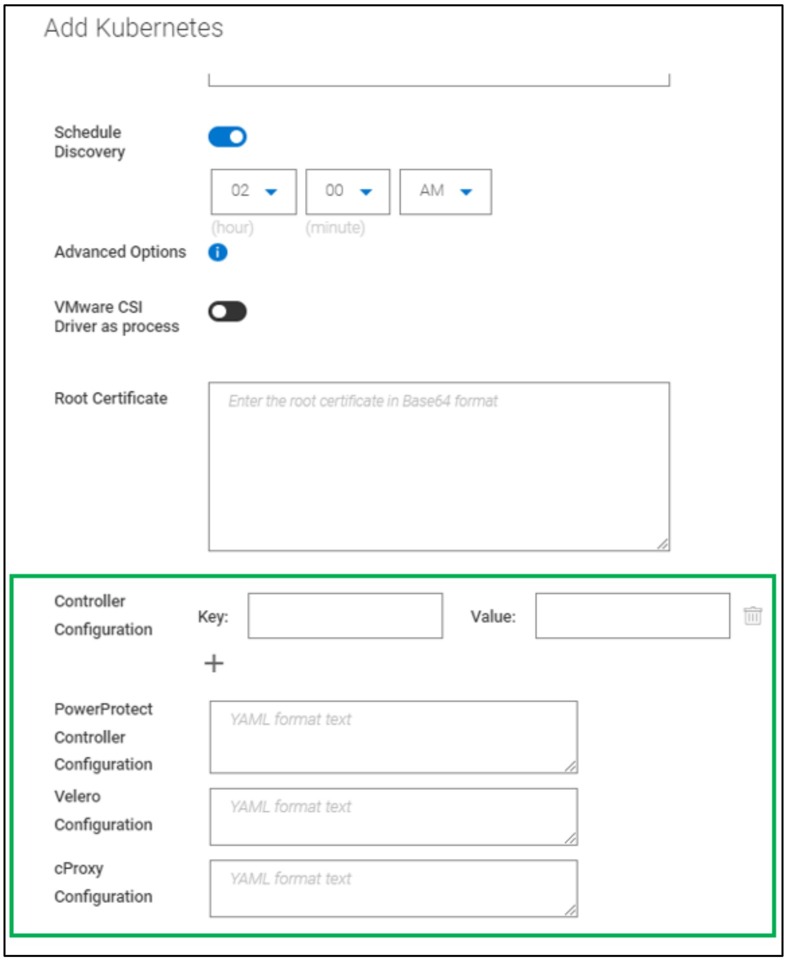
4.1.4 Controller Configurations

Within the PowerProtect Data Manager UI, you can add controller configurations for a Kubernetes cluster.

When adding Network Interface Cards (NICs), setting DNS configuration for pods, or creating custom ports, you might want to update the PowerProtect Controller, Velero, and cProxy pod configurations to apply additional attributes or change existing attributes.

When adding the Kubernetes cluster as an asset source, PowerProtect Data Manager UI provides the ability to update the PowerProtect Controller configuration, Velero configuration, or cProxy configuration fields, which can be used to add NICs or set the DNS configuration for pods.

Pod information is specified in "**Advanced Options**" when adding or editing the Kubernetes cluster asset source in the PowerProtect Data Manager UI.



Add Kubernetes

Schedule Discovery: on

02 (hour) 00 (minute) AM

Advanced Options

VMware CSI Driver as process: off

Root Certificate: Enter the root certificate in Base64 format

Controller Configuration — Key: Value:

PowerProtect Controller Configuration: YAML format text

Velero Configuration: YAML format text

cProxy Configuration: YAML format text

See the controller configuration section in PowerProtect Data Manager Kubernetes User Guide for more information.

4.2 Volume group snapshotting for Kubernetes PVCs

The PowerFlex CSI driver volumegroup snapshot extension enables you to snapshot all Kubernetes PVCs that belong to the same volume group, instead of creating a snapshot of each PVC using the existing CSI volumesnapshot functionality.

PowerProtect Data Manager automatically uses the volumegroup snapshot extension when the following conditions are met:

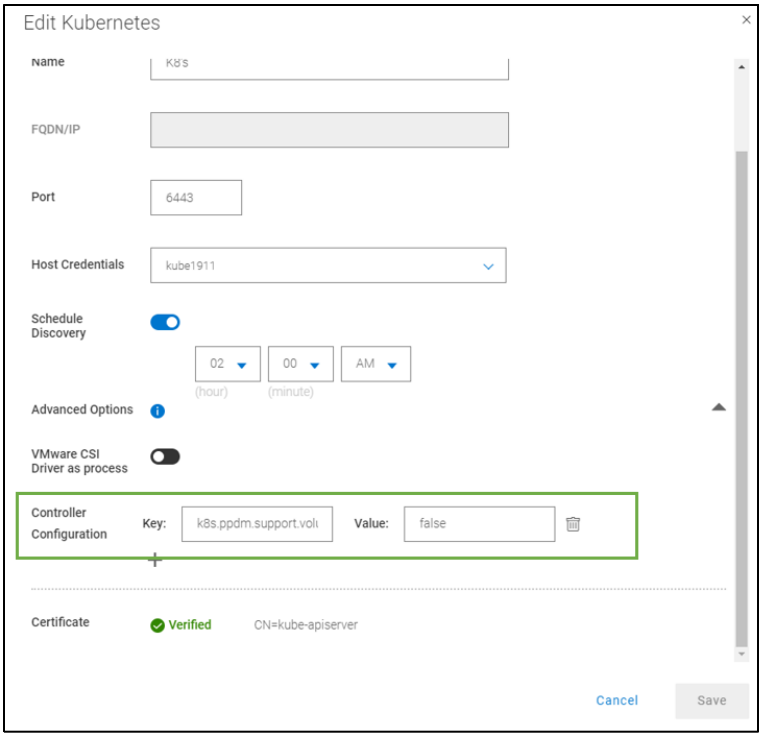
- The Kubernetes clusters are using PVCs provisioned by the CSI driver for PowerFlex
- The CSI driver for PowerFlex has the volumegroup snapshotter feature enabled, and the volumegroupsnapshot CRD is present on the Kubernetes cluster
- The PVCs share the same volume-group label

When the volumegroup snapshotter feature is in use for a group of PVCs in a protection policy, an entry for

VolumeGroup appears in the Details pane of the PowerProtect Data Manager UI Jobs window for the protection policy backup.

If you want PowerProtect Data Manager to use the volumesnapshot functionality instead of the PowerFlex volumegroup snapshot extension, you can disable the volumegroup snapshot extension by performing the following steps:

1. From the PowerProtect Data Manager UI, go to Infrastructure > Asset Sources.
2. In the Kubernetes tab, select the Kubernetes cluster asset source that is used to protect the PVCs belonging to the volumegroup, and then click Edit.
3. In the Edit Kubernetes dialog box, click the down arrow to expand Advanced Settings.
4. Under Controller Configuration, set the value for the property “k8s.ppdm.support.volumeGroup” to false.



Edit Kubernetes

Name: K8's

FQDN/IP:

Port: 6443

Host Credentials: kube1911

Schedule Discovery: 02 (hour) 00 (minute) AM

Advanced Options

VMware CSI Driver as process

Controller Configuration — Key: k8s.ppdm.support.volu Value: false

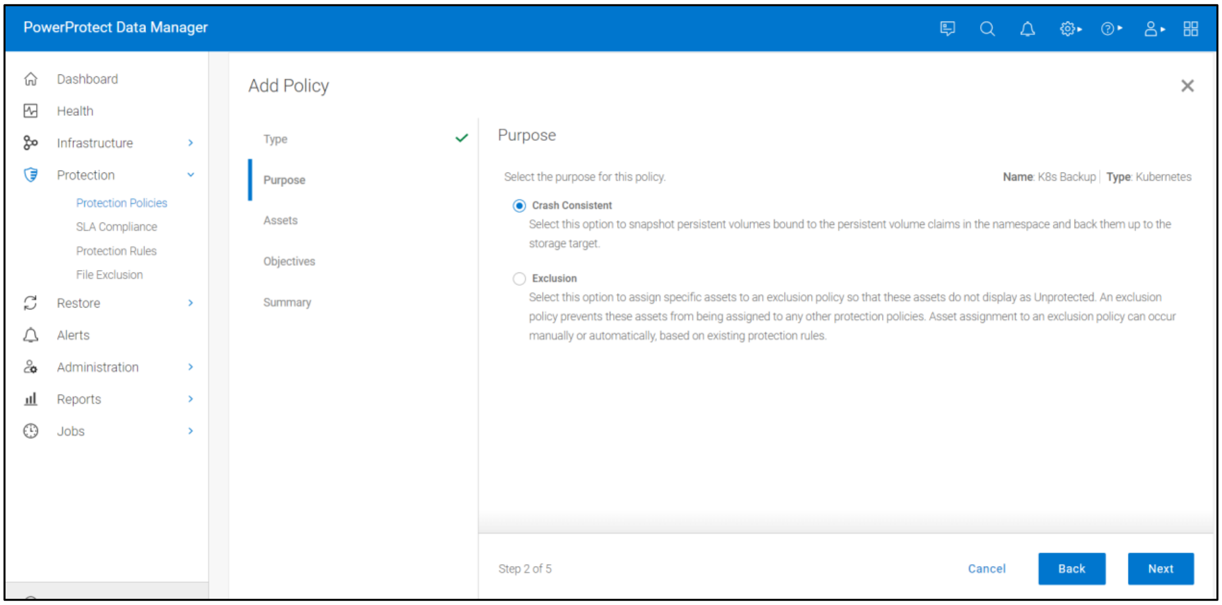
Certificate: Verified CN=kube-apiserver

4.3 Add a protection policy for Kubernetes namespace protection

Protection policies define a set of objectives that apply to specific periods of time. These objectives drive configuration, active protection, and copy-data-management operations that satisfy the business requirements for the specified data. Each plan type has its own set of user objectives. Users with the system admin role can create protection policies. A Kubernetes protection policy enables you to select namespaces in the Kubernetes cluster that you want to back up. Use the Data Manager UI to create a Kubernetes namespace protection policy.

Data Manager provides the following options when creating a Kubernetes cluster protection policy:

- **Crash Consistent:** Select this type for point-in-time backup of namespaces.
- **Exclusion:** Select this type if there are assets within the protection policy to exclude from data protection operations.



Purpose

Select the purpose for this policy.

Name: K8s Backup | **Type**: Kubernetes

● **Crash Consistent**
Select this option to snapshot persistent volumes bound to the persistent volume claims in the namespace and back them up to the storage target.

○ **Exclusion**
Select this option to assign specific assets to an exclusion policy so that these assets do not display as Unprotected. An exclusion policy prevents these assets from being assigned to any other protection policies. Asset assignment to an exclusion policy can occur manually or automatically, based on existing protection rules.

Also, the admin can select namespaces and associated PVCs statically or dynamically for inclusion or exclusion in protection policies, along with schedules, retention, and other protection operations.

From the Jobs window, the progress of the new Kubernetes cluster protection policy backup and associated tasks can be monitored.

For more information about creating a protection policy, see the section “Add a protection policy for Kubernetes namespace protection” in the PowerProtect Data Manager Kubernetes User Guide.

4.4 Protecting PVCs in PowerScale access zones

Starting with version 19.11, PowerProtect Data Manager supports the protection of PVCs provisioned in different PowerScale access zones.

PowerProtect Data Manager by default creates a cProxy pod in the powerprotect namespace when backing up and restoring PVCs to a new namespace. The powerprotect namespace may not have access to all the PowerScale access zones.

This feature is useful in scenarios in which PowerProtect Data Manager is unable to protect PVCs by default, such as:

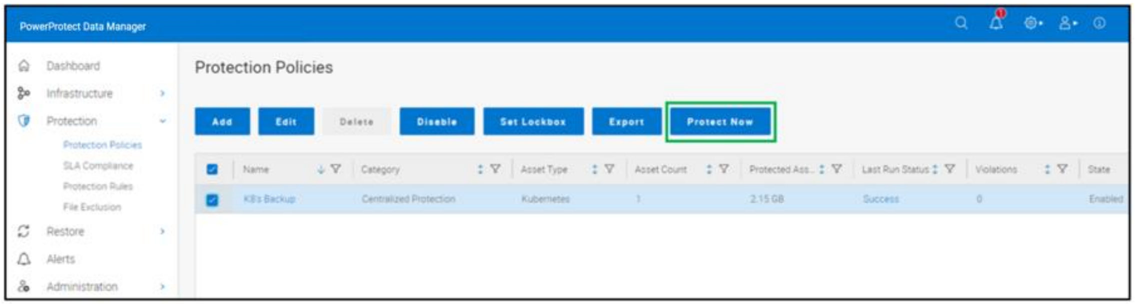
- When PVCs from multiple access zones are provisioned in the Kubernetes cluster
- When Kubernetes cluster firewall and networking are configured not to allow PowerProtect Data Manager data mover pods running in the powerprotect namespace access to PVCs from all access zones

For more information about configuring this feature to protect PVCs in PowerScale access zones, see the section “Protecting PVCs in PowerScale access zones” in the PowerProtect Data Manager Kubernetes User Guide.

5 Performing a backup of namespaces and PVCs on Kubernetes cluster

Apart from the scheduled backup option, PowerProtect Data Manager supports the option to perform manual backups.

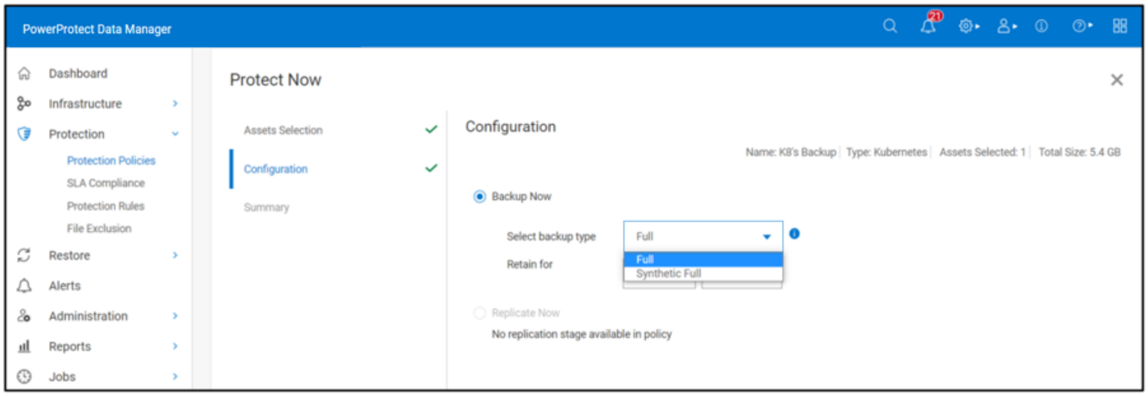
To perform a manual backup, go to **Protection > Protection Policies**. Select the Kubernetes protection policy and click **Protect Now** as shown here.



Protection Policies

☑	Name	Category	Asset Type	Asset Count	Protected Ass...	Last Run Status	Violations	State
☑	K8's Backup	Centralized Protection	Kubernetes	1	2.15 GB	Success	0	Enabled

During the manual backup, select the backup type of either **Full** or **Synthetic Full**.



Protect Now

Configuration

Name: K8's Backup | Type: Kubernetes | Assets Selected: 1 | Total Size: 5.4 GB

● Backup Now

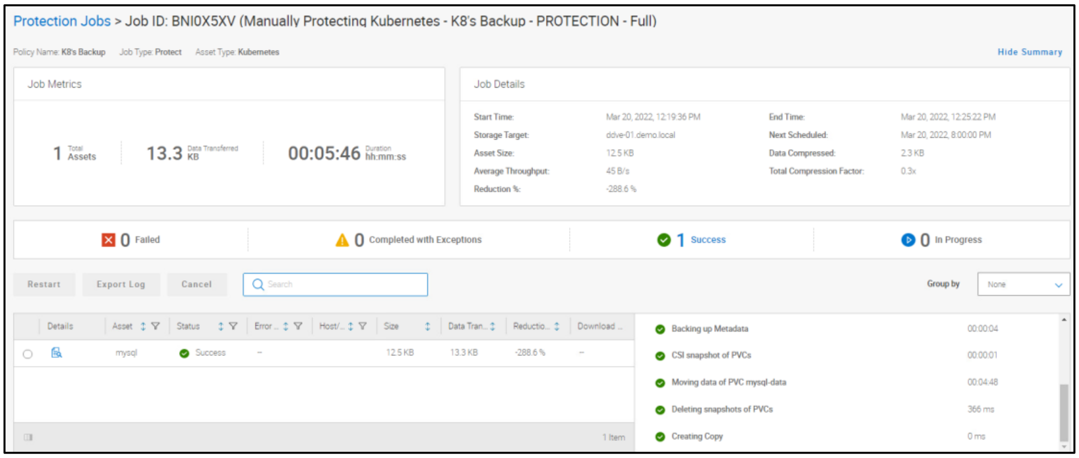
Select backup type: Full

Retain for

○ Replicate Now

No replication stage available in policy

From the **Jobs** > **Protection Jobs** window, you can monitor the progress of the Kubernetes cluster backup.



Protection Jobs > Job ID: BNI0X5XV (Manually Protecting Kubernetes - K8's Backup - PROTECTION - Full)

Policy Name: K8's Backup Job Type: Protect Asset Type: Kubernetes

Job Metrics

1 Total Assets | 13.3 KB Data Transferred | 00:05:46 Duration hh:mm:ss

Job Details

Start Time:	Mar 20, 2022, 12:19:36 PM	End Time:	Mar 20, 2022, 12:25:22 PM
Storage Target:	ddve-01.demo.local	Next Scheduled:	Mar 20, 2022, 8:00:00 PM
Asset Size:	12.5 KB	Data Compressed:	2.3 KB
Average Throughput:	45 B/s	Total Compression Factor:	0.3x
Reduction %:	-288.6 %		

☒ 0 Failed | ⚠ 0 Completed with Exceptions | ✓ 1 Success | ▶ 0 In Progress

	Details	Asset	Status	Error ...	Host/...	Size	Data Tran...	Reductio...	Download ...
○		mysql	✓ Success	--		12.5 KB	13.3 KB	-288.6 %	--

✓ Backing up Metadata	00:00:04
✓ CSI snapshot of PVCs	00:00:01
✓ Moving data of PVC mysql-data	00:04:48
✓ Deleting snapshots of PVCs	366 ms
✓ Creating Copy	0 ms

For more information about performing backup and recovery of Kubernetes workloads using PowerProtect Data Manager, see the PowerProtect Data Manager Kubernetes User Guide.

6 Application-consistent database backups in Kubernetes

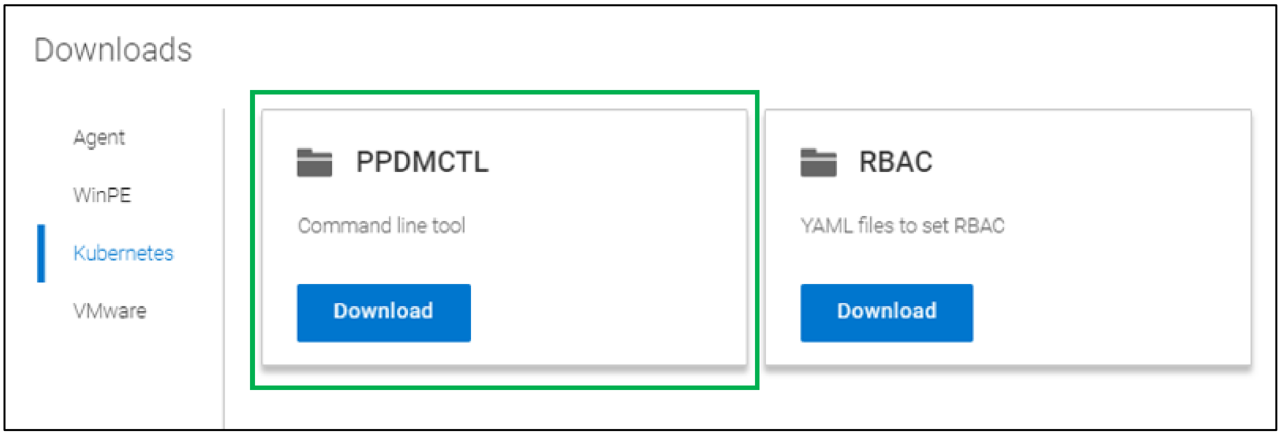
PowerProtect Data Manager supports agentless, application-consistent backups of database applications that reside in Kubernetes pods. Application-consistent backups occur when the database application is informed of a pending backup. The database completes all pending transactions and operations, while typically queuing new requests. This process places the database in a quiescent state of relative inactivity where the backup represents a true snapshot of the application. This backup now captures items that would have otherwise been stored only in memory. After the snapshot, the application resumes normal functionality. In most environments, the snapshot operation is instantaneous, so downtime is minimal.

These backups are agentless, in that the PowerProtect Data Manager can take a snapshot of containers without the need for software installation in the database application environment. Then, that snapshot is backed up using the normal procedures for the Kubernetes environment.

The PowerProtect Data Manager provides a standardized way to quiesce a supported database, back up the data from that database, and then return the database to operation. Application templates serve as a bridge between a specific database environment and the Kubernetes backup architecture for the PowerProtect Data Manager. Depending on the differences between database environments, each deployment may require a different configuration file.

Application templates are typically deployed from customizable YAML files that come with the CLI package. The CLI package exists on the PowerProtect Data Manager host at `/usr/local/brs/lib/cndm/misc/ppdmctl.tar.gz` and is part of the PowerProtect Data Manager deployment. Starting from PowerProtect Data Manager 19.12, the CLI package can be downloaded from the PowerProtect Data Manager UI.

1. In the PowerProtect Data Manager UI, click the **System Settings** icon and then select **Downloads**.
2. In the left pane, select **Kubernetes**.
3. In the **PPDMCTL** box, click **Download**.



6.1.1 Supported database applications

Supported applications include:

- **MySQL** in the following configurations:
  - Standalone deployment in one pod.
  - Cluster (primary/secondary) deployment with multiple StatefulSets or ReplicaSets. For example, through Helm.
- **MongoDB** without shards.
- **PostgreSQL** in the following configurations:
  - Standalone deployment in one pod.
  - Cluster (primary/secondary) deployment with multiple StatefulSets. For example, through Helm.
- **Cassandra** without shards.

Because data syncs from the primary pods to secondary pods, the PowerProtect Data Manager backs up secondary pods first.

6.1.2 Kubernetes MySQL Application Consistent protection

Data Manager has Application Consistent protection feature for MySQL and MongoDB databases. To use this feature, user creates a MySQL Application consistent backup using templates. PowerProtect Data Manager has in-built ‘ppdmctl.tar.gz’ file available which contains the templates for MySQL and MongoDB in YAML format. This file is pushed from PowerProtect Data Manager to Kubernetes cluster root directory to enable the application-consistent protection. The protection is provided only to the namespace which has asset defined in the protection policy. When the protection policy is applied with an enabled MySQL template, the consistency of the backed-up data is application consistent; when the MySQL template is disabled, the consistency is crash consistent.

Kubernetes namespaces are already discovered as assets with PowerProtect Data Manager. There are the available namespaces for protection except powerprotect and velero-ppdm namespaces. To configure the application-consistent protection, particular namespace is specified. In this case, test-namespace is used to demonstrate.

1. Log in to the **PowerProtect Data Manager CLI** with administrator credentials.
2. Change the directory to **cndm/misc** using the following command.

   ```
   cd /usr/local/brs/lib/cndm/misc
   ```

3. Run the list command (ls) to view the content which is ppdmctl.tar.gz file.

   ```
   ls -ltrh
   ```

4. Run the following command to transfer file from PowerProtect Data Manager to Kubernetes cluster (K8s-cluster01 is for reference for Kubernetes cluster.

   ```
   scp ppdmctl.tar.gz tme@k8s-cluster01
   ```

5. Log in to Kubernetes Cluster, run the list command to list the content of the root directory, and confirm **ppdmctl.tar.gz** is available.
6. Run the tar command to untar the file. Once untar is completed, the ppdmctl directory is created.

```
tar -xvf ppdmctl.tar.gz
```

7. Run the change directory command to enter the ppdmctl directory, and run ls to list the content.

```
cd ppdmctl
ls
```

8. Run the following command to enter the examples directory where all the MySQL and MongoDB templates are stored.

```
cd examples
ls
```

9. Copy the `mysqlapptemplatehelm.yaml` file to create a template copy from the example for MySQL with command (`tmedemomysqlapptemplatestadalone.yaml` is the existing template associated to the test- namespace).

```
cp mysqlapptemplatehelm.yaml tmedemomysqlapptemplatestadalone.yaml
```

10. Run the following command to describe the MySQL stateful set.

```
kubectl describe sts mysql -n test-namespace
```

11. Edit the tmedemomysqlapptemplatestadalone.yaml file to make the changes.

    - To change the name of primary pod as per the pod description, edit: `selectorExpression: “mysql”` (mysql is new name)
    - If there are no worker pods being used for MySQL, remove the `selectorTerms` section under `selectors` for worker.
    - To edit the MySQL password environment variable, preHook and postHook are edited. (Pod-based hooks execute hook code in a new pod derived from template in a deployment configuration, preHook is to quiesce and postHook is to be unquiesce.)

12. Run `cd..` to exit the examples directory to the ppdmctl directory.
13. To create a template for MySQL application-consistent protection from the edited (step11) template file in test-namespace, use the ppdmctl utility by running the following command.

```
./ppdmctl template create tmemysqltemplate –type=mysql –namespace=test-
namespace –inputfile=examples/tmedemomysqlapptemplatestadalone.yaml
```

The application-consistent protection is configured for specific namespace (test-namespace in this demonstration). When the backup is run manually or using protection policy, the backup copy is saved as application consistent at the target storage.

14. To disable the MySQL template, run the following command.

```
./ppdmctl template disable tmemysqltemplate –namespace=test-namespace
```

6.1.3 Kubernetes PostgreSQL Application Consistent Protection

PowerProtect Data Manager supports both PostgreSQL and PostgreSQL HA (high availability). Both PostgreSQL and PostgreSQL HA configure a cluster with a primary-standby topology. The primary node has writing permission while replication is on the standby nodes which have read-only permissions. PowerProtect Data Manager has the integrated **ppdmctl.tar.gz** file which contains the templates for PostgreSQL in YAML format. This file is pushed to the primary node. Pgpool-II acts as proxy for PostgreSQL backend. It reduces connection overhead and used as a load balancer for PostgreSQL. Pgpool-II is responsible to spread the queries among nodes.

6.1.1 Application template of PostgreSQL

```
AppLabel: “app=postgresql”
Type: “Postgresql”
AppActions:
Pod:
PreHook: "[\"/bin/bash\",  \"-c\",     \"PGPASSWORD=$POSTGRES_PASSWORD psql  - U
  $POSTGRES_USER -c \\\"select pg_start_backup('ppdm-backup', true);\\\"\"]"
PostHook: "[\"/bin/bash\", \"-c\", \"PGPASSWORD=$POSTGRES_PASSWORD psql -  U
  $POSTGRES_USER -c \\\"select pg_stop_backup();\\\"\"]"
```

Application template of PostgreSQL HA:

```
AppLabel:“app.kubernetes.io/name=postgresql-
ha,app.kubernetes.io/component=postgresql”

# applabel to select
pod Type: “Postgresql”
AppActions:
Pod:
PreHook: "[\"/bin/bash\", \"-c\", \"REPMGR_PRIMARY_HOST=`echo
$REPMGR_PRIMARY_HOST | cut -f1 -d '.'`; if [ $HOSTNAME =
$REPMGR_PRIMARY_HOST ]; then PGPASSWORD=$POSTGRES_PASSWORD psql -U
$POSTGRES_USER -c \\\"select     pg_start_backup('ppdm-backup',
                                 true);\\\"; fi\"]“
PostHook: "[\"/bin/bash\", \"-c\", \"REPMGR_PRIMARY_HOST=`echo
      $REPMGR_PRIMARY_HOST | cut -f1 -d '.'`; if [ $HOSTNAME =
$REPMGR_PRIMARY_HOST ]; then PGPASSWORD=$POSTGRES_PASSWORD psql -U
$POSTGRES_USER -c \\\"select     pg_stop_backup();\\\"; fi\"]"
Application:
Kind: StatefulSet
Selectors:
SelectorTerms: {“field” : “Name”, "selectorExpression": ".*-[1-9][0- 9]*$“ }
# Standby pods with index > 0
SelectorTerms: {“field” : “Name”, "selectorExpression": ".*-0$"} # Primary pods
with index = 0
```

6.1.2 Kubernetes Cassandra application-consistent protection

Apache Cassandra is highly scalable, high-performance distributed database designed to handle large amounts of data with no single point of failure. It is a type of NoSQL database. Cassandra partitions data over storage nodes using consistent hashing algorithm. Each node stores multiple ranges of tokens and each range of token replicates to multiple nodes for fault-tolerance and high availability. PowerProtect Data Manager has Application Consistent protection feature for Cassandra database. PowerProtect Data Manager has the integrated **ppdmctl.tar.gz** file which contains the templates for Cassandra in YAML format.

Application template of Cassandra:

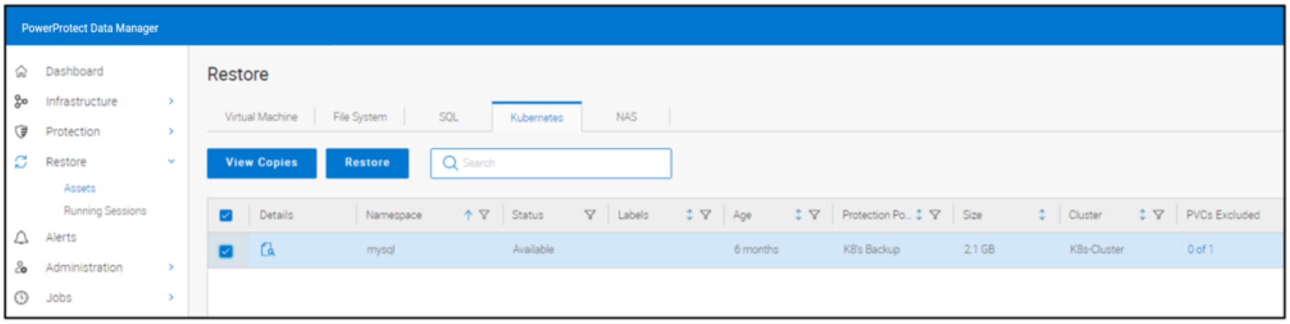
```
AppLabel: “app=cassandra”
Type:  “Cassandra”
Enable: true
AppActions:
Pod:
PreHook: "[\"/bin/bash\", \"-c\", \"nodetool flush\"]"
```

For more details, see the section “Application-Consistent Database Backups in Kubernetes“ in the PowerProtect Data Manager Kubernetes User Guide.

7 Restoring Kubernetes namespaces and PVCs

After protecting the Kubernetes cluster protection policy, restoring namespace and PVCs can be done from individual namespace backups. When a protection policy is successfully backed up, PowerProtect Data Manager displays details such as the name of the storage system containing the asset backup, location, the creation and expiry date, and the size.

To view backup copies available for restore, select **Restore > Assets** on the PowerProtect Data Manager UI. Select the asset and click **Restore**.



☑	Details	Namespace	Status	Labels	Age	Protection Po...	Size	Cluster	PVCs Excluded
☑		mysql	Available		6 months	K8's Backup	2.1 GB	K8s-Cluster	0 of 1

PowerProtect Data Manager provides options to recover the Kubernetes namespaces to the same or to an alternate cluster.

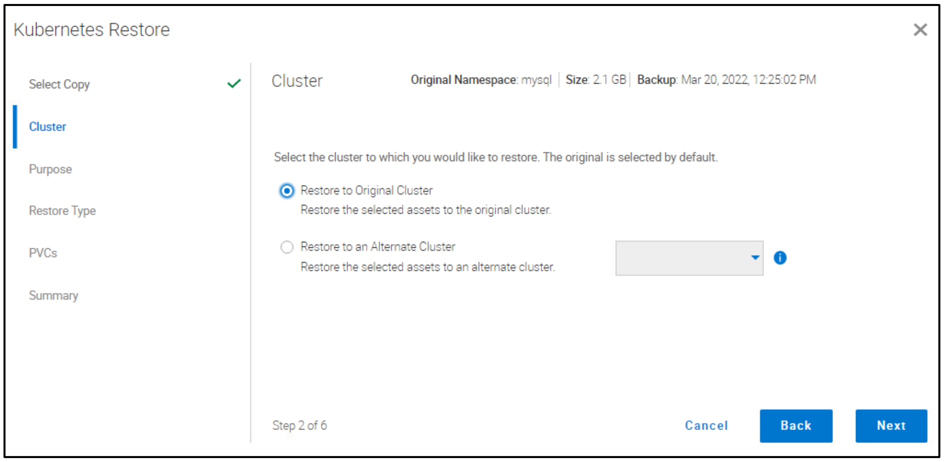
Restore to Original Cluster: Select this option to restore to a new namespace on the original cluster.

Restore to an Alternate Cluster: Select this option to restore to a new namespace on a different cluster, and then select the cluster from the list.

A restore to an alternate cluster can be useful when:

- Migrating namespaces from a cluster on-premises to a cluster in the cloud.
- Moving namespaces from a lower cluster version to a higher cluster version.
- Moving from one environment to another (for example, from a test environment to a production environment).

Note: When restoring to an alternate cluster, ensure that this Kubernetes cluster has been added and discovered in the PowerProtect Data Manager UI Asset Sources window.



Cluster

Original Namespace: mysql | **Size**: 2.1 GB | **Backup**: Mar 20, 2022, 12:25:02 PM

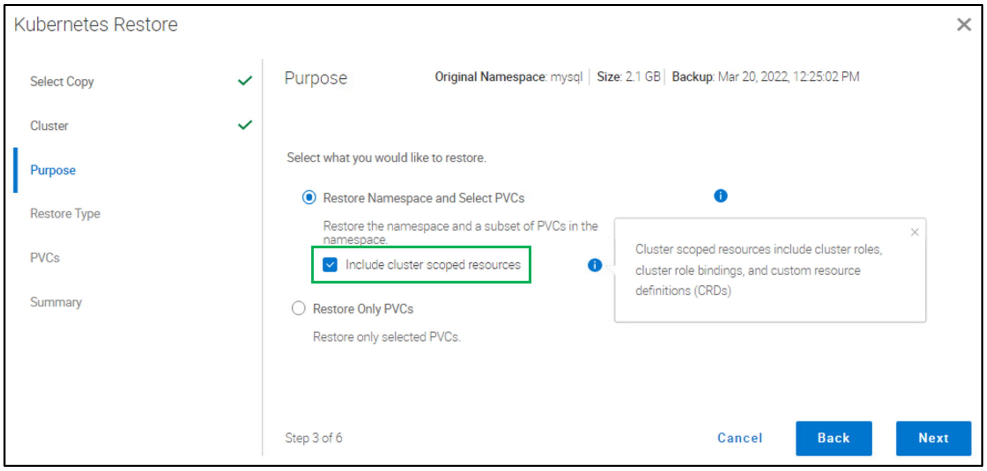
Select the cluster to which you would like to restore. The original is selected by default.

● Restore to Original Cluster
Restore the selected assets to the original cluster.

○ Restore to an Alternate Cluster
Restore the selected assets to an alternate cluster.

Step 2 of 6

Restore Namespace and Select PVCs: Select this option to restore namespace resources and selected persistent volume claims (PVCs). Optionally, you can also select **Include cluster scoped resources** to restore the cluster roles, cluster role bindings, and custom resource definitions (CRDs) that were backed up automatically as part of the Kubernetes protection policy. This option is only available for PowerProtect Data Manager 19.6 and later Kubernetes protection policy backups.



Kubernetes Restore

Select Copy ✓
Cluster ✓
Purpose
Restore Type
PVCs
Summary

Purpose — **Original Namespace**: mysql | **Size**: 2.1 GB | **Backup**: Mar 20, 2022, 12:25:02 PM

Select what you would like to restore.

● Restore Namespace and Select PVCs

Restore the namespace and a subset of PVCs in the namespace.

☑ Include cluster scoped resources

Cluster scoped resources include cluster roles, cluster role bindings, and custom resource definitions (CRDs)

○ Restore Only PVCs

Restore only selected PVCs.

Step 3 of 6

Restore Cluster Scoped Resources:

The resources that are scoped at a cluster level and not bound to any specific namespace are called cluster scoped resources (for example, cluster roles, cluster role bindings, and custom resource definitions (CRD)). When the CRD is created, Kubernetes API server creates a new RESTful resource path for the specific version created. The CRD can be either namespaced or cluster scoped as specified in the scope field. This section examines how you can use Kubernetes backup copies and restore the cluster scoped resources such as service accounts, cluster roles, and cluster bindings. The Velero component performs the backup and restore of cluster scoped resources, including the backup of the namespace and the associated cluster scoped resources, such as cluster roles and cluster role bindings. Custom resource definitions are included as a part of each namespace backup.

Restoring cluster resources is controlled by a check-box option in UI named **Include cluster scoped resources** and runs the restore process.

Below are the steps to verify the cluster role, cluster binding, and CRD.

1. Log in to Kubernetes cluster and run the following command to view cluster role.

   ```
   kubectl get clusterrole
   ```

2. Run the following command to view the clusterrolebinding.

   ```
   kubectl get clusterrolebinding
   ```

3. Run the following command to view the cluster resource definitions.

   ```
   kubectl get crd
   ```

4. Run the following command to view the service account associated with the namespace.

   ```
   kubectl get serviceaccount -n <namespace>
   ```

5. To delete all the cluster scoped resources associated with the namespace, run the

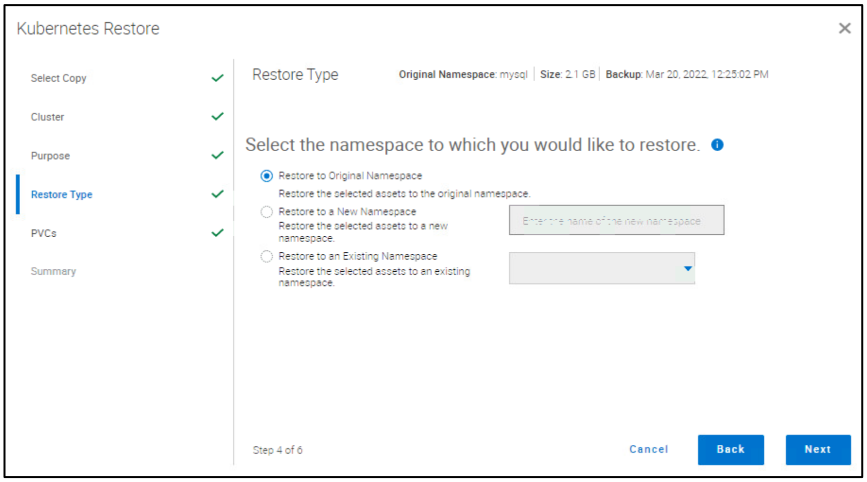
below script on Kubernetes cluster.

```
s./del-fcd.sh
```

After protecting the Kubernetes cluster protection policy, restoring namespace and PVCs can be done from individual namespace backups.

Use the following recovery options:

- **Restore to original namespace**: Restore to the original namespace on the original cluster.
- **Restore to new namespace**: Create a namespace and restore to this location on the original cluster or a different cluster.
- **Restore to existing namespace**: Restore to an existing namespace in the original cluster or a different cluster.



Kubernetes Restore

Restore Type **Original Namespace**: mysql | **Size**: 2.1 GB | **Backup**: Mar 20, 2022, 12:25:02 PM

Select the namespace to which you would like to restore.

● Restore to Original Namespace
Restore the selected assets to the original namespace.

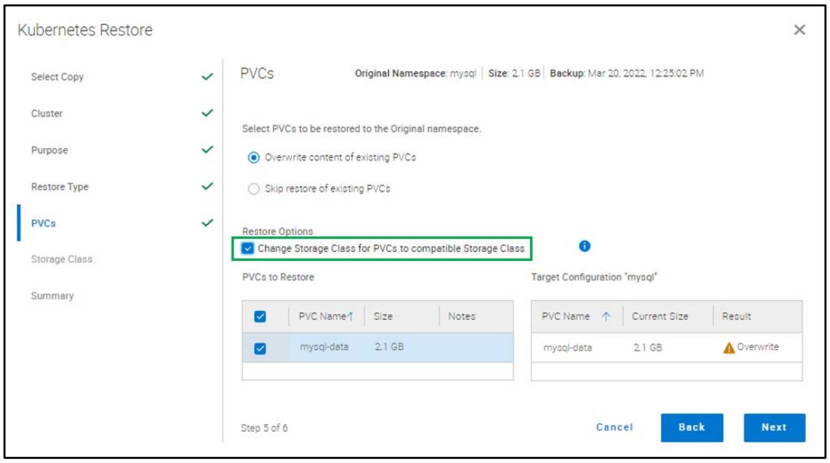
○ Restore to a New Namespace
Restore the selected assets to a new namespace.
Enter the name of the new namespace

○ Restore to an Existing Namespace
Restore the selected assets to an existing namespace.

Step 4 of 6

On the PVCs page, if the configuration of the namespace you want to restore is different from the configuration in the target namespace, perform the following:

- Select **Overwrite content of existing PVCs** to overwrite existing PVCs in the target location with the PVCs being restored if the PVCs have the same name.
- Select **Skip restore of existing PVCs** to restore selected PVCs without overwriting existing PVCs in the target location if they have the same name.



Kubernetes Restore

PVCs **Original Namespace**: mysql | **Size**: 2.1 GB | **Backup**: Mar 20, 2022, 12:25:02 PM

Select PVCs to be restored to the Original namespace.

● Overwrite content of existing PVCs

○ Skip restore of existing PVCs

Restore Options

☑ Change Storage Class for PVCs to compatible Storage Class

PVCs to Restore

☑	PVC Name↑	Size	Notes
☑	mysql-data	2.1 GB	

Target Configuration "mysql"

PVC Name ↑	Current Size	Result
mysql-data	2.1 GB	⚠ Overwrite

Step 5 of 6

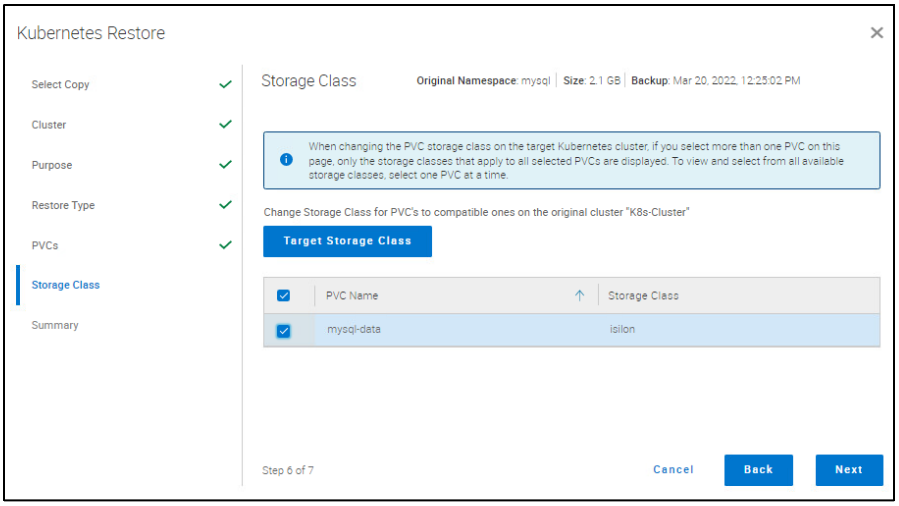
Optionally, if you choose to retire the storage class on the original cluster, perform the following

- Select **Change storage class for PVCs to compatible storage class**. The PVCs that are part of the restore display.
- Select the check box next to the PVCs for which you want to change the storage class on the target cluster.

The storage class mapping feature with PowerProtect Data Manager 19.8 enables you to choose an alternate storage class for PVCs with a certain provisioner type while restoring persistent volumes. Storage class mapping enables restoring namespaces and PVCs from one cluster to another using different container storage. It is also useful when the migration of data from one storage class to another storage class and from on-premises to cloud or conversely.

Note: The storage class is not modified for existing PVCs being overwritten.

If **Change storage class for PVCs to compatible storage class** is selected as shown above, the Storage Class page appears with a list of supported storage classes on the target cluster.



Kubernetes Restore

Storage Class   **Original Namespace**: mysql | **Size**: 2.1 GB | **Backup**: Mar 20, 2022, 12:25:02 PM

When changing the PVC storage class on the target Kubernetes cluster, if you select more than one PVC on this page, only the storage classes that apply to all selected PVCs are displayed. To view and select from all available storage classes, select one PVC at a time.

Change Storage Class for PVC's to compatible ones on the original cluster "K8s-Cluster"

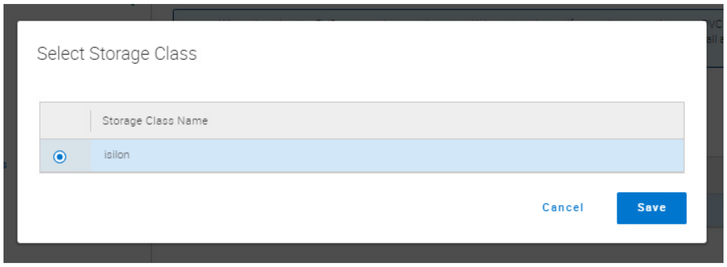
☑	PVC Name ↑	Storage Class
☑	mysql-data	isilon

Step 6 of 7

1. Select the check box next to a PVC for which you want to change the storage class on the target cluster. Alternately, select multiple PVCs to change all selections to the same storage class.

Note: When changing the PVC storage class on the target Kubernetes cluster, if you select more than one PVC at a time on this page, only the storage classes that apply to all selected PVCs are displayed. To view and select from all available storage classes, select one PVC at a time.

2. Click **Target Storage Class** to select from the available storage classes. The **Select Storage Class** dialog appears.

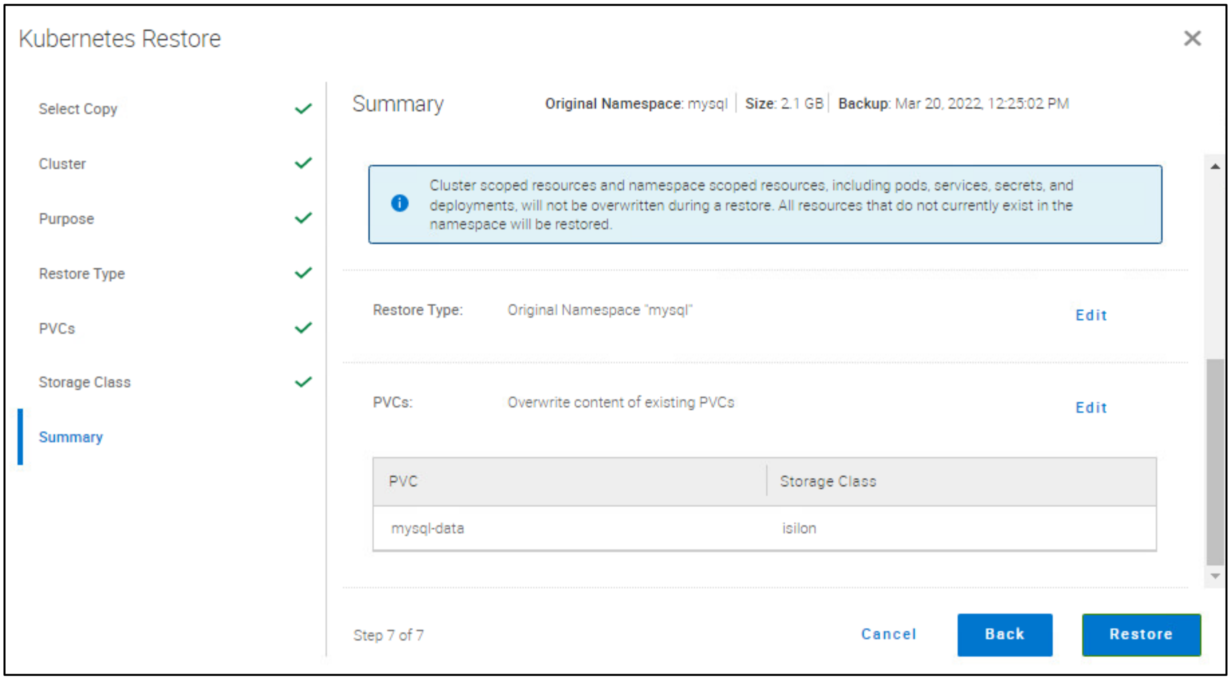


Select Storage Class

	Storage Class Name
●	isilon

3. Select from one of the available storage classes, and click **Save** to save your changes and return to the **Storage Class** page.

From the **Summary** page, click **Restore** to initiate the restore job. An informational dialog box appears indicating that the restore has started.


Kubernetes Restore

Summary **Original Namespace**: mysql | **Size**: 2.1 GB | **Backup**: Mar 20, 2022, 12:25:02 PM

Cluster scoped resources and namespace scoped resources, including pods, services, secrets, and deployments, will not be overwritten during a restore. All resources that do not currently exist in the namespace will be restored.

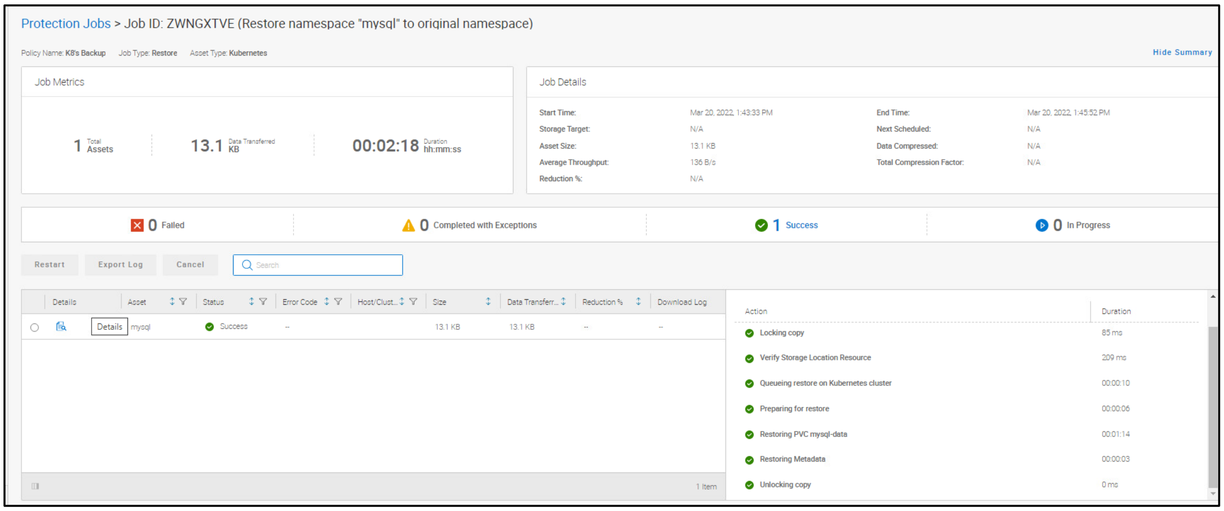
Restore Type: Original Namespace "mysql"

PVCs: Overwrite content of existing PVCs

PVC	Storage Class
mysql-data	isilon

Step 7 of 7

From the **Jobs** > **Protection Jobs** window, the restore progress can be monitored.


Protection Jobs > Job ID: ZWNGXTVE (Restore namespace "mysql" to original namespace)

Policy Name: K8's Backup   Job Type: Restore   Asset Type: Kubernetes

Job Metrics

1 Total Assets   13.1 KB Data Transferred   00:02:18 Duration hh:mm:ss

Job Details

Start Time:	Mar 20, 2022, 1:43:33 PM	End Time:	Mar 20, 2022, 1:45:52 PM
Storage Target:	N/A	Next Scheduled:	N/A
Asset Size:	13.1 KB	Data Compressed:	N/A
Average Throughput:	136 B/s	Total Compression Factor:	N/A
Reduction %:	N/A		

0 Failed   0 Completed with Exceptions   1 Success   0 In Progress

	Details	Asset	Status	Error Code	Host/Clust...	Size	Data Transferr...	Reduction %	Download Log
○	Details	mysql	Success	--		13.1 KB	13.1 KB	--	--

1 Item

Action	Duration
Locking copy	85 ms
Verify Storage Location Resource	209 ms
Queueing restore on Kubernetes cluster	00:00:10
Preparing for restore	00:00:06
Restoring PVC mysql-data	00:01:14
Restoring Metadata	00:00:03
Unlocking copy	0 ms

7.1 Quick recovery to an alternate Kubernetes cluster

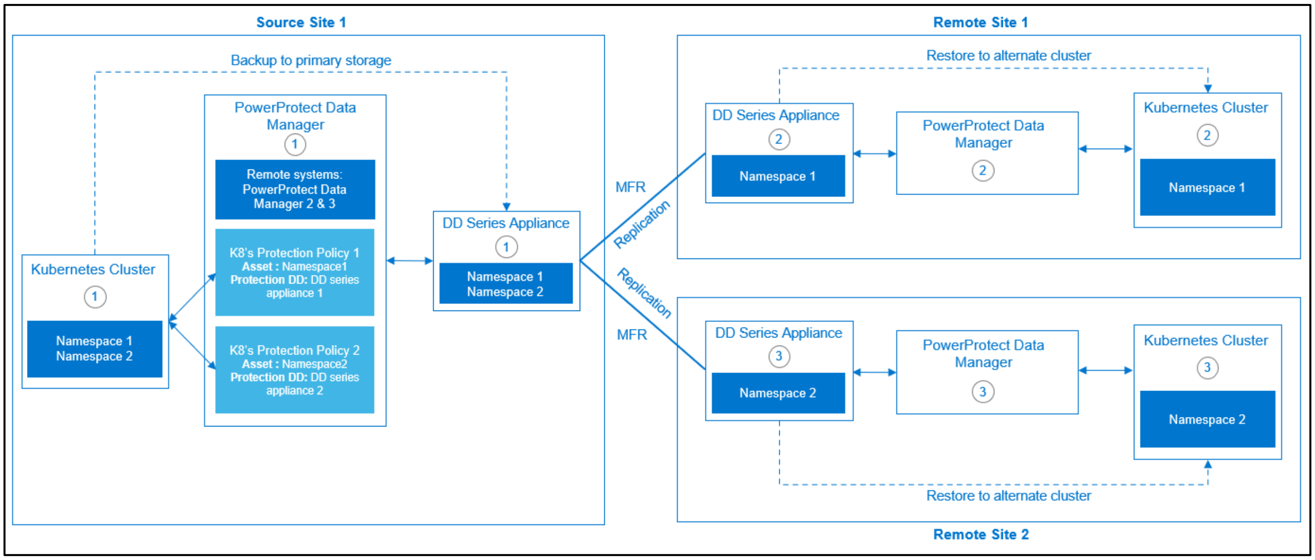
The quick recovery feature enables you to restore assets and data that you replicated to a destination system at a remote site. Quick recovery is supported for the protected assets of the following PowerProtect Data Manager asset sources:

- Virtual machines
- Kubernetes
- File system
- NAS

Quick recovery sends metadata from the source system to the destination system, following the same flow of backup copies. This metadata makes the replication destination aware of the copies and enables the recovery view. You can recover your workloads at the remote site before you restore the source PowerProtect Data Manager system.

7.1.1 PowerProtect Data Manager quick recovery for Kubernetes cluster

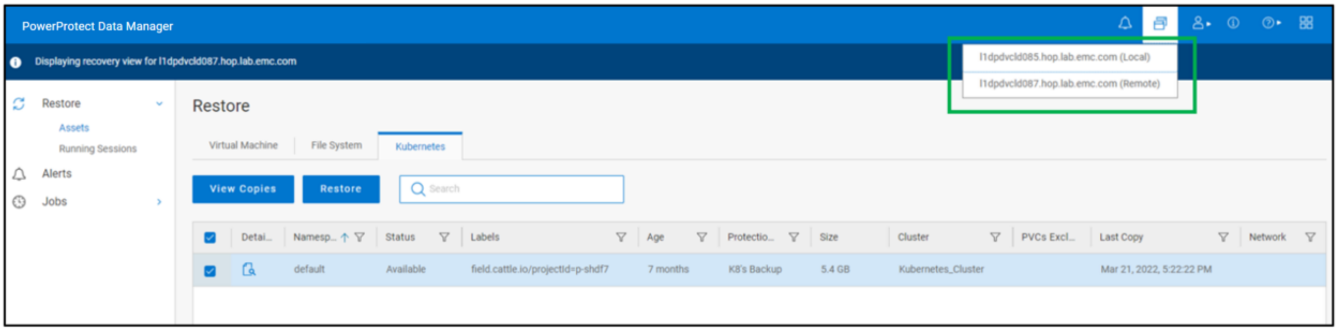
Starting with version 19.10, PowerProtect Data Manager supports quick recovery to an alternate Kubernetes cluster for Kubernetes workloads.



On a quick-recovery-enabled environment with a source and remote PowerProtect Data Manager configuration, Kubernetes assets that have protected and replicated copies created in the source PowerProtect Data Manager will be synced to the remote PowerProtect Data Manager.

The replicated copies will be available for restore to a Kubernetes cluster that is added to the remote PowerProtect Data Manager.

Remote PowerProtect Data Manager displaying the assets of the source PowerProtect Data Manager for restore:

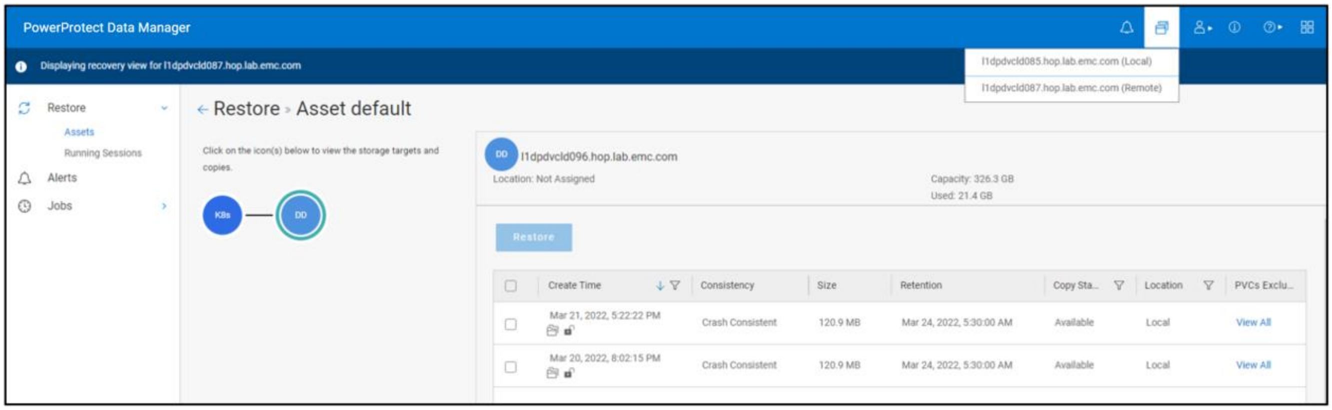


Displaying recovery view for l1dpdvcld087.hop.lab.emc.com

Restore

☑	Detai...	Namesp...	Status	Labels	Age	Protectio...	Size	Cluster	PVCs Excl...	Last Copy	Network
☑		default	Available	field.cattle.io/projectId=p-shdf7	7 months	K8's Backup	5.4 GB	Kubernetes_Cluster		Mar 21, 2022, 5:22:22 PM	

Remote PowerProtect Data Manager displaying the replicated backup copies of the source PowerProtect Data Manager asset for restore:


PowerProtect Data Manager
l1dpdvcld085.hop.lab.emc.com (Local)
l1dpdvcld087.hop.lab.emc.com (Remote)
Restore
Assets
Running Sessions
Alerts
Jobs
K8s
DD
DD
Restore

Displaying recovery view for l1dpdvcld087.hop.lab.emc.com

← Restore › Asset default

Click on the icon(s) below to view the storage targets and copies.

l1dpdvcld096.hop.lab.emc.com

Location: Not Assigned

Capacity: 326.3 GB

Used: 21.4 GB

□	Create Time	Consistency	Size	Retention	Copy Sta...	Location	PVCs Exclu...
□	Mar 21, 2022, 5:22:22 PM	Crash Consistent	120.9 MB	Mar 24, 2022, 5:30:00 AM	Available	Local	View All
□	Mar 20, 2022, 8:02:15 PM	Crash Consistent	120.9 MB	Mar 24, 2022, 5:30:00 AM	Available	Local	View All

A Technical support and resources

Dell.com/support is focused on meeting customer needs with proven services and support.

The Data Protection Info Hub provides expertise that helps to ensure customer success with Dell data protection products.

A.1 Related resources

Dell PowerProtect Data Manager:

- PowerProtect Data Manager Kubernetes User Guide
- PowerProtect Data Manager Administration and User Guide
- PowerProtect Data Manager compatibility matrix