

# Dell PowerStore: VMware vSphere with Tanzu and TKG Clusters

May 2024

H18731.4

## White Paper

### Abstract

This document describes the configuration and integration between Dell PowerStore arrays and VMware vSphere with Tanzu and Tanzu Kubernetes Grid guest clusters.

## Copyright

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.

Use, copying, and distribution of any software described in this publication requires an applicable software license.

Copyright © 2021–2024 Dell Inc. or its subsidiaries. All Rights Reserved. Published in the USA May 2024 H18731.4.

Dell Inc. believes the information in this document is accurate as of its publication date. The information is subject to change without notice.

# Contents

Executive summary.....4

Introduction .....5

Setup prerequisites.....8

Preparing and configuring PowerStore storage .....10

Preparing and configuring vSphere and PowerStore .....19

Deploying vSphere with Tanzu.....26

Day 2 Kubernetes .....30

vSphere with Tanzu and Metro Volume witness .....52

VMware Tanzu Kubernetes Grid.....54

Conclusion.....55

Bibliography .....55

Appendix: Additional resources .....56

# Executive summary

## Overview

Containers and container management are changing the way applications and services are developed and delivered to businesses globally. In most instances, automated deployment, scaling, management, and orchestration of containerized applications becomes a necessity. Kubernetes is an open-source orchestrator for deploying containerized applications. It was originally developed by Google, inspired by a decade of experience deploying scalable, reliable systems in containers by using application-oriented APIs. (Burns, Beda, & Hightower, 2019)

VMware vSphere with Tanzu transforms a vSphere cluster into developer-ready infrastructure with the ability to run Kubernetes workloads managed by a Kubernetes control plane in the hypervisor. Dell PowerStore unified storage arrays support the block, file, and vVol storage types used by vSphere with Tanzu and VMware Tanzu Kubernetes Grid Service (TKG) clusters. Combining vSphere with Tanzu and Dell PowerStore is a good fit for modernized applications.

## Audience

This document is intended for IT administrators, storage architects, partners, and Dell Technologies employees with presumed knowledge of vSphere with Tanzu. This audience also includes individuals who may evaluate, acquire, manage, operate, or design a Dell networked storage environment using PowerStore systems.

## Revisions

Date	Part number/ revision	Description
April 2021	H18731	Initial release
April 2022	H18731.1	Updated template
May 2023	H18731.2	Updated for PowerStoreOS 3.5 (secure snapshots)
October 2023	H18731.3	Updated for PowerStoreOS 3.6
May 2024	H18731.4	Updated for PowerStoreOS 4.0 Removed references to PowerStore X

## We value your feedback

Dell Technologies and the authors of this document welcome your feedback on this document. Contact the Dell Technologies team by [Email](#).

**Author:** Jason Boche

---

**Note:** For links to other documentation for this topic, see the [PowerStore Info Hub](#).

---

# Introduction

## White paper overview

This paper provides configuration examples, tips, recommended settings, and other storage guidelines to follow while integrating VMware vSphere with Tanzu and TKG clusters with Dell PowerStore.

There are various links to technical resources at the end of this document. We recommend becoming familiar with and using these resources when deploying vSphere with Tanzu.

## PowerStore overview

PowerStore achieves new levels of operational simplicity and agility. It uses a container-based microservices architecture, advanced storage technologies, and integrated machine learning to unlock the power of your data. PowerStore is a versatile platform with a performance-centric design that delivers multidimensional scale, always-on data reduction, and support for next-generation media.

PowerStore brings the simplicity of public cloud to on-premises infrastructure, streamlining operations with an integrated machine-learning engine and seamless automation. It also offers predictive analytics to easily monitor, analyze, and troubleshoot the environment. PowerStore is highly adaptable, providing the flexibility to host specialized workloads directly on the appliance and modernize infrastructure without disruption. It also offers investment protection through flexible payment solutions and data-in-place upgrades.

The PowerStore platform is available as a bare-metal, unified storage array that can service block, file, and VMware vSphere Virtual Volumes (vVols) resources along with numerous data services and efficiencies.

## Terminology

The following table provides definitions for some of the terms that are used in this document.

**Table 1. Terminology**

Term	Definition
Appliance	Solution containing a base enclosure and attached expansion enclosures. The size of an appliance could be only the base enclosure or the base enclosure plus expansion enclosures.
Cloud Native Storage (CNS)	A vCenter Server component that manages persistent volumes.
Container	Software that packages code and dependencies making it portable to run across computing environments.
Container Image Virtual Disk	A local cache of images that can be used in pods instead of pulling down images from an external container registry
Ephemeral Virtual Disk	Transient storage for vSphere Pods. Each vSphere Pod has one ephemeral disk that exists throughout the life cycle of the vSphere Pod. Ephemeral disks persist across container restarts but are removed when the vSphere Pod is removed.
Fibre Channel (FC) protocol	Protocol used to perform IP and SCSI commands over a Fibre Channel network.
File system	Storage resource that can be accessed through file-sharing protocols such as SMB or NFS.

Term	Definition
First Class Disk (FCD)	Also known as an Improved Virtual Disk (IVD). A type of virtual disk used to back persistent volumes. It is unique in that it is designed to offer operational and life cycle management without being associated with a virtual machine.
Internal snapshot (replication snapshot)	Unified snapshots created by the system that are part of an asynchronous replication session. These snapshots are only visible in the PowerStore CLI or PowerStore REST API, and manual modification is not possible. Each asynchronous replication session uses up to two internal snapshots that are taken on the source and destination storage resources. Each session also takes up one read/write snapshot on the destination storage system. The last successful internal read-only (RO) snapshots for source and destination storage resources and are used as a common base.
iSCSI	Provides a mechanism for accessing block-level data storage over network connections.
kubectl	The official Kubernetes client. A command-line tool for interacting with the Kubernetes API.
Network-attached storage (NAS) server	File-level storage server used to host file systems. A NAS server is required to create file systems that use SMB or NFS shares.
Network File System (NFS)	An access protocol that allows data access from Linux or UNIX hosts on a network.
Persistent Volume	Storage resource that is used to maintain stateful data for containerized applications. A static persistent volume is a virtual disk (.vmdk) created in advance by the vSphere administrator. A dynamic persistent volume is provisioned on demand using a storage class. Both types are consumed by a Persistent Volume Claim (PVC). Persistent volumes maintain stateful data by having a life cycle independent of the containerized application that uses it. When the consuming pod is deleted, the persistent volume remains.
Persistent Volume Claim	A request for static or dynamic persistent storage. A persistent volume claim (PVC) is tied to a persistent volume.
PowerStore base enclosure	Enclosure containing both nodes (node A and node B) and 25 NVMe drive slots
PowerStore cluster	Multiple appliances in a single grouping. Clusters can consist of one appliance or more. Up to four PowerStore appliances can be clustered by adding appliances as required.
PowerStore Command Line Interface (PSTCLI)	Tool which can be installed on an operating system to manage a PowerStore system. It allows a user to perform tasks on the storage system by typing commands instead of using the user interface.
PowerStore expansion enclosure	Enclosures that can be attached to a base enclosure to provide additional storage.
PowerStore Manager	An HTML5 management interface for creating storage resources and configuring and scheduling protection of stored data on PowerStore. PowerStore Manager can be used for all management of PowerStore native replication.

Term	Definition
PowerStore node	Storage controller that provides the processing resources for performing storage operations and servicing I/O between storage and hosts. Each PowerStore appliance contains two nodes.
PowerStore Representational State Transfer (REST) API	Set of resources (objects), operations, and attributes that provide interactive, scripted, and programmatic management control of the PowerStore cluster.
PowerStore Q model	Container-based storage system that is running on purpose-built hardware. This storage system supports unified (block and file) workloads, or block-optimized workloads. The PowerStore Q model supports Quad-Level Cell (QLC) NVMe SSDs for data storage.
PowerStore T model	Container-based storage system that is running on purpose-built hardware. This storage system supports unified (block and file) workloads, or block-optimized workloads. The PowerStore T model supports Triple-Level Cell (TLC) NVMe SSDs for data storage.
Snapshot	Also called a unified snapshot, a snapshot is a point-in-time view of a storage resource or data stored on a storage resource. A user can recover files from a snapshot, restore a storage resource from a snapshot, or provide snapshot data access to a host. When a snapshot is taken, it creates an exact copy of the source storage resource and shares all blocks of data with it. As data changes on the source, new blocks are allocated and written to. Unified snapshot technology can be used to take a snapshot of a block or file storage resource.
Storage class	A type of storage available to and used to back a persistent volume. A storage class abstracts underlying storage constructs. A storage class is used to describe storage requirements for a persistent volume. Used by DevOps teams in persistent volume claims.
Storage resource	Top-level object that a user can provision which is associated with a specific quantity of storage. All host access and data-protection activities are performed at this level. In this document, storage resources refer to resources that support replication such as volumes, volume groups, and thin clones.
Tanzu Kubernetes Cluster	Deployed on a vSphere with Tanzu Supervisor Cluster by using the Tanzu Kubernetes Grid Service, it is a full distribution of the open-source Kubernetes container orchestration platform. Tanzu Kubernetes clusters use the open-source Photon operating system from VMware and are built, signed, and supported by VMware.
Thin clone	Read/write copy of a thin block storage resource (volume, volume group, or VMware vSphere VMFS datastore) that shares blocks with the parent resource.
User snapshot	A snapshot that is created manually by the user or by a protection policy with an associated snapshot rule. This snapshot type is different than an internal snapshot, which the system takes automatically using asynchronous replication.

Term	Definition
Virtual Volumes (vVols)	VMware storage framework which allows VM data to be stored on individual Virtual Volumes. This ability allows data services to be applied at a VM-granularity level while using Storage Policy Based Management (SPBM).
Volume	A block-level storage device that can be shared out using a protocol such as iSCSI or Fibre Channel. It represents a SCSI logical unit.
Volume group	Storage instance which contains one or more volumes within a storage system. Volume groups can be configured with write-order consistency and help organize the storage that is allocated for particular hosts.
vStorage API for Array Integration (VAAI)	VMware API that allows storage-related tasks to be offloaded to the storage system.
vSphere API for Storage Awareness (VASA)	VMware API that provides additional insight about the storage capabilities in vSphere.
vSphere Namespace	Organizational unit or abstraction onto which vSphere administrators apply policies and assign to development teams. Typically contains vSphere Pods or a Tanzu Kubernetes cluster. Backed by a Resource Pool to control resource limits.
vSphere Pod	A virtual machine with a small footprint that runs one or more Linux containers. Right sized with explicit resource reservations for the workload that it accommodates at deployment. Equivalent to a Kubernetes pod.
vSphere with Tanzu	vSphere infrastructure with workload management enabled making it a platform for running Kubernetes workloads natively on the hypervisor layer.
Workload	In vSphere with Tanzu, workloads are applications deployed as standard virtual machines, containers running inside vSphere Pods, Tanzu Kubernetes clusters, or applications that run inside Tanzu Kubernetes clusters.

## Setup prerequisites

### Introduction

The focus of this document is to cover the integration and use of vSphere with Tanzu on PowerStore. Deploying and configuring vSphere with Tanzu has its own setup prerequisites, largely networking, and is covered in depth in the VMware document [vSphere with Tanzu Configuration and Management](#). Use that as a guide to deploy vSphere with Tanzu in your environment.

Before deploying vSphere with Tanzu, a few infrastructure components will need to be in place.

### PowerStore



vSphere Pods and TKG clusters require storage for control plane nodes, ephemeral virtual disks, container image virtual disks, TKG clusters, content library, and persistent volumes. The vSphere Container Storage Interface (CSI) driver allows vSphere with Kubernetes and TKG cluster to consume VMFS, NFS, vVol, and vSAN storage.

- If vSphere with Tanzu will be using VMFS or vVol storage only, PowerStore can be deployed in block optimized mode.
- If vSphere with Tanzu will be using NFS storage, deploy PowerStore in unified mode.

Become familiar with the following documents and use them as a guide when deploying and configuring PowerStore for VMware vSphere environments:

- [Dell PowerStore: Best Practices Guide](#)
- [Dell PowerStore: Virtualization Integration](#)
- [Dell PowerStore Virtualization Infrastructure Guide](#)

## VMware vSphere and networking

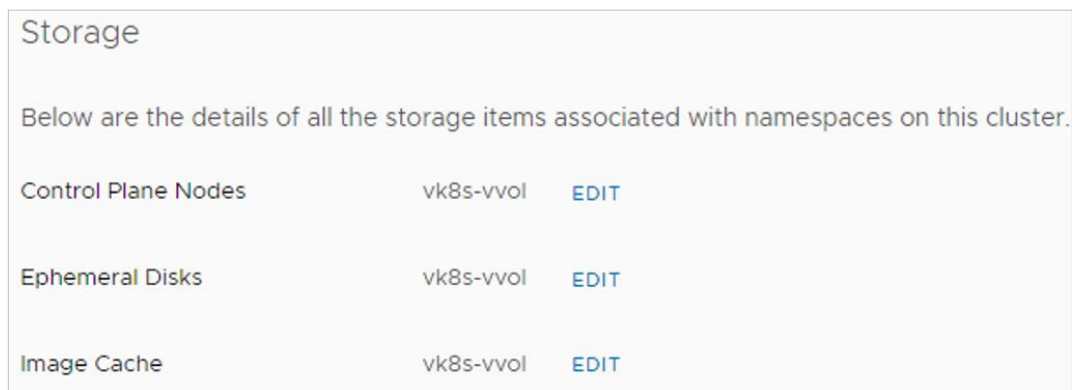
Before deploying vSphere with Tanzu, a vSphere 7 cluster must be deployed. vSphere with Tanzu requires a minimum of three ESXi hosts with HA and DRS enabled. In addition, the DRS Automation Level must be configured for Fully Automated. vSphere DRS determines the placement of control plane VMs on ESXi hosts and can migrate them as needed. vSphere DRS is also integrated with the Kubernetes Scheduler on the control plane VMs, so that DRS determines the placement of vSphere Pods. vSphere Pod placement requests first go through the regular Kubernetes workflow, and then to DRS, which makes the final placement decision. A Supervisor Cluster can either use the vSphere networking stack with the NSX Advanced Load Balancer (also known as Avi Load Balancer, Essentials Edition) or an open-source load balancer such as HAProxy or VMware NSX-T Data Center to provide connectivity to Kubernetes control plane VMs, services, and workloads. Lastly, the cluster should be zoned or configured to consume PowerStore storage following documented best practices from both VMware and Dell.

## Tanzu edition license

A Tanzu edition license enables the workload management functionality in vSphere 7.0.1. After deploying vSphere with Tanzu, a Tanzu edition license needs to be installed before the 60-day evaluation period expires. When the evaluation period expires, some features become unavailable. Examples include creating namespaces, updating the Kubernetes version of the Supervisor Cluster, and inability to deploy new vSphere Pods and Tanzu Kubernetes clusters.

## Policies and tags

Using the vSphere CSI driver, vSphere with Tanzu and TKG clusters can consume various shared storage types for various roles. Storage is assigned in the vSphere UI and .yaml files using a storage class which correlates to policies and tags in vSphere. Policies and tags can be added, modified, and deleted as needed throughout the life cycle of vSphere with Tanzu. However, the initial deployment of vSphere with Tanzu does require storage assignment for the supervisor control plane nodes, pod ephemeral disks, and container image cache. At least one VM Storage Policy must be configured before deploying vSphere with Tanzu. Configuring policies and tags with PowerStore will be covered in an upcoming section.



**Figure 1.** Example of vVol storage used for vSphere with Tanzu deployment

## Preparing and configuring PowerStore storage

### Overview

Using the vSphere CSI driver, cloud native applications (CNA) deployed in containers and managed by Kubernetes can consume various storage types provided by PowerStore. The following sections will provide guidance and examples for each storage type.

Storage can be added throughout the vSphere with Tanzu life cycle. However, at least one volume must be configured before vSphere with Tanzu can be deployed to support the placement of the Supervisor Control Plane Nodes, Ephemeral Disks, and Image Cache. Storage is also needed for the Content Library. The storage designation for each of these can be changed later after deployment.

See the following table for infrastructure volume sizing guidelines. Keep in mind that volumes can be expanded as needed.

**Table 2.** Compute and storage resource requirements for vSphere with Tanzu

Virtual machine	Nodes	Total vCPUs	Total memory	Total storage
Supervisor Cluster control plane (small nodes – up to 2,000 pods per Supervisor cluster)	3	12	48 GB	200 GB *
Registry Service	N/A	7	7 GB	200 GB
Tanzu Kubernetes Cluster control plane (small nodes)	3 (per cluster)	6	12 GB	48 GB
Tanzu Kubernetes Cluster worker nodes (small nodes)	3 (per cluster)	6	12 GB	48 GB

Virtual machine	Nodes	Total vCPUs	Total memory	Total storage
VMware NSX-T Edge node **	2	16	16	400 GB

\* Source: [Sizing Compute and Storage Resources for a vSphere with Tanzu Workload Domain](#)

\*\* If using NSX-T for networking and load balancer

**Note:** When using block or file storage, consider deploying multiple datastores for the Supervisor Cluster Control Plane Node VMs. VMware recommends using Storage DRS with anti-affinity rules to maintain Control Plane Node VMs on separate datastores. Additional information about this topic is provided in an upcoming section of this document.

PowerStore automatically manages the underlying storage for maximum performance and capacity, eliminating the need for administrators to configure RAID or storage pools. In the following storage provisioning examples, manually configuring these options is unnecessary in PowerStore.

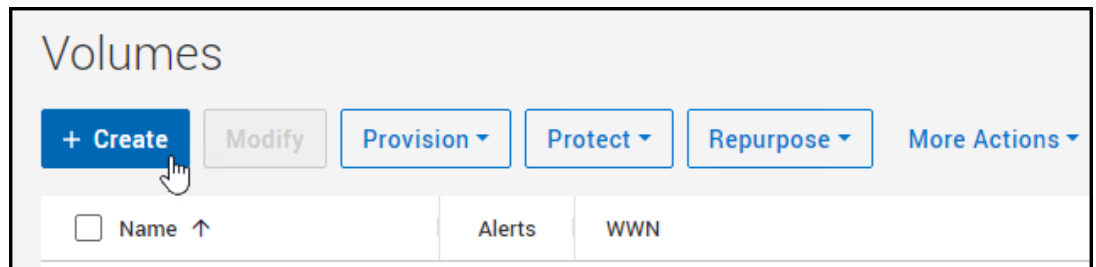
## Block volumes

Block volume storage can be used for Supervisor Control Plane Nodes, Ephemeral Disks, Image Cache, Content Library, and Persistent Volumes. Once deployed, block volumes are discovered by vSphere and formatted with the VMFS file system. The resulting datastores are tagged and associated with a VM Storage Policy so that they can be consumed by vSphere with Tanzu or Tanzu Kubernetes Clusters.

**Note:** If PowerStore will only be serving block volumes, consider installing PowerStore in block-optimized mode, which disables the NAS and vVol capabilities. This mode can increase the scale or performance of PowerStore since it can devote additional CPU and memory that is no longer needed for file.

To provision a block volume in PowerStore Manager, follow these steps:

1. From the Storage | Volumes menu, click **Create**.



2. Specify name, quantity, and size.

Create Volumes

**Properties**

Host Mappings

Summary

**General**

Name (Or Prefix)  
ps-12vk8s\_15

Description - optional

Category  
Virtualization

Application  
Containers/Kubernetes

Quantity  
1

Size  
200 GB

**Additional Properties**

Associated Volume Group - optional  
None Selected [Select](#)

Volume Protection Policy - optional  
None

QoS Policy - optional  
None

Volume Performance Policy  
Medium

The volume performance policy determines the relative performance priority of the volume using a share-based system. Available options are Low, Medium (default), and High. The volume performance policy can be tied to a tag and a VM Storage Policy and used as a storage tiering mechanism. For example,

- a. Volume with a High-volume performance policy can be tied to a Gold VM Storage Policy.
  - b. Volume with a Medium volume performance policy can be tied to a Silver VM Storage Policy.
  - c. Volume with a Low volume performance policy can be tied to a Bronze VM Storage Policy.
3. Choose the vSphere cluster object or individual host objects to map the block volume to. A LUN number can be manually specified if wanted.

### Host Mappings

**Available Hosts/Host Groups** ⓘ

Select the hosts or host groups based on storage protocol to be mapped to the volume.

☒ SCSI (Supports iSCSI or FC transport layer)  
☐ NVMe (Supports NVMe TCP and FC transport layers)

12 Hosts & Host Groups, 1 selected

Name	OS	Host/Host Group	Initiator Type	vSphere Host Name
s1262.techsol.local_fc	ESXi	Host	FC	--
s1262.techsol.local_iscsi	ESXi	Host	iSCSI	--
s1263.techsol.local_fc	ESXi	Host	FC	--
s1263.techsol.local_iscsi	ESXi	Host	iSCSI	--
<input checked="" type="checkbox"/> Tanzu Cluster	ESXi	Host Group	iSCSI	--

### Logical Unit Number

The logical unit number is used to help the host identify the volume to which they are mapped. Provide a specific logical unit number or have it generated.

☐ Generate Automatically ☒ Provide a Logical Unit Number

Logical Unit Number

15

#### 4. Review the information on the summary page.

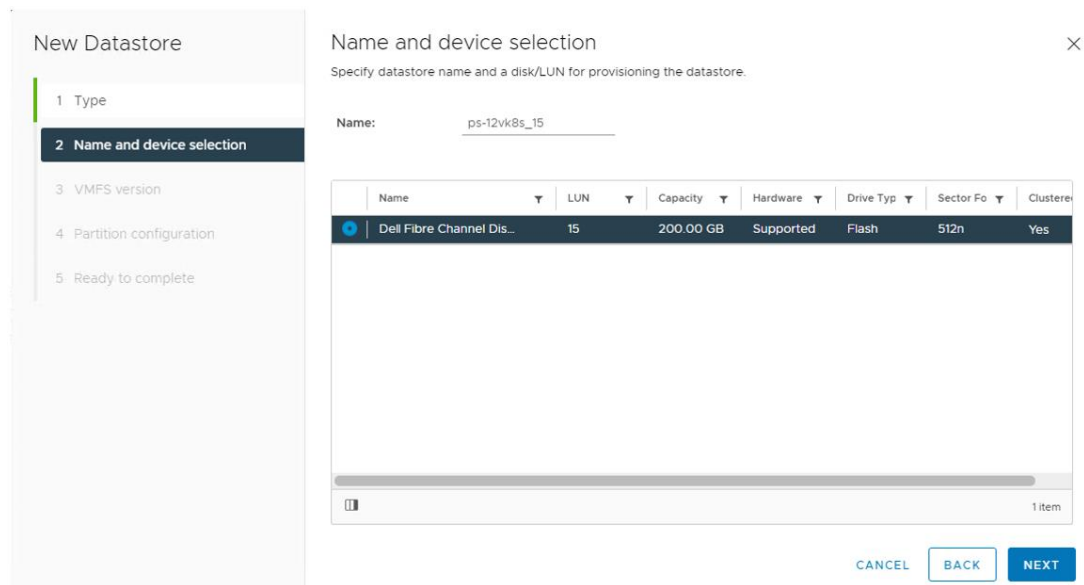
### Create Volumes

Properties ✓  
Host Mappings ✓  
**Summary**

#### Summary

Total Size	200.0 GB	The following volumes will be created: • ps-12vk8s_15
Category	Virtualization	
Application	Containers/Kubernetes	
Placement	PS-3-appliance-1	
Hosts/Host Groups	1	
Volume Group	--	
Performance Policy	Medium	
QoS Policy		
Protection Policy	--	

#### 5. In the vSphere UI, rescan for storage and follow the New Datastore workflow to consume the volume and format it with VMFS.



## Volume groups and protection policies

A volume group is a logical container for a group of block volumes and provides a single point of management for multiple storage resources that work together as a unit. For example, use volume groups to monitor metrics. This would be a useful application for vSphere with Tanzu.



**Figure 2. Monitoring performance a volume group consisting of three volume members**

Volume groups can also be used to consistently apply snapshot and replication data protection policies across all volumes in the volume group. These features are beneficial for traditional storage hosts and application types. However, many of the objects introduced in vSphere with Tanzu such as Supervisor Cluster Control Plane Nodes, Pods, and Tanzu Kubernetes Clusters cannot be directly managed from the vSphere UI. In addition, some of these objects are not visible as traditional storage object types within PowerStore Manager and they do not contain stateful user or application data. For these vSphere with Tanzu object types, it is not practical to use traditional PowerStore Manager snapshot and asynchronous or synchronous replication for data protection. However, one exception might be Persistent Volumes. Persistent Volumes are backed by independent First Class Disks (FCD) which may contain stateful data that should be protected. In this use case, array-based snapshots and/or asynchronous/synchronous/Metro Volume replication might be a good fit. For more information about protecting VMware Tanzu Clusters, refer to the section later in this document covering Metro Volume as well as the document [Dell EMC PowerProtect Data Manager protecting VMware Tanzu Kubernetes Clusters](#).

## Secure snapshots

Starting with PowerStoreOS 3.5, the secure snapshot setting can be enabled for snapshots on volumes and volume groups. Snapshot rules can also be configured to create secure snapshots automatically. With secure snapshots enabled, the snapshots and parent resource are protected from accidental or malicious deletion and serve as a cost-effective line of defense against ransomware attacks. If an unauthorized user gains access to a system, the attacker cannot delete secure snapshots and cause data loss.

**Figure 3. Workflow to create a snapshot of a volume or volume group**

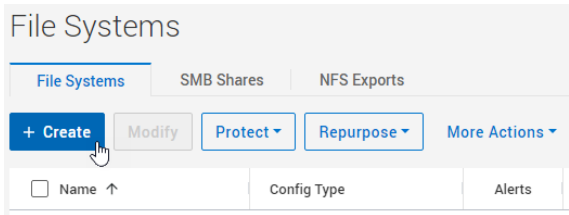
## NFS volumes

PowerStore File Systems and NFS Exports can be consumed as NFS datastores in vSphere. NFS datastores can be used for Supervisor Control Plane Nodes, Ephemeral Disks, Image Cache, Content Library, and Persistent Volumes. Like block volumes, NFS datastores are tagged and associated with a VM Storage Policy so that they can be consumed by vSphere with Tanzu or Tanzu Kubernetes Clusters.

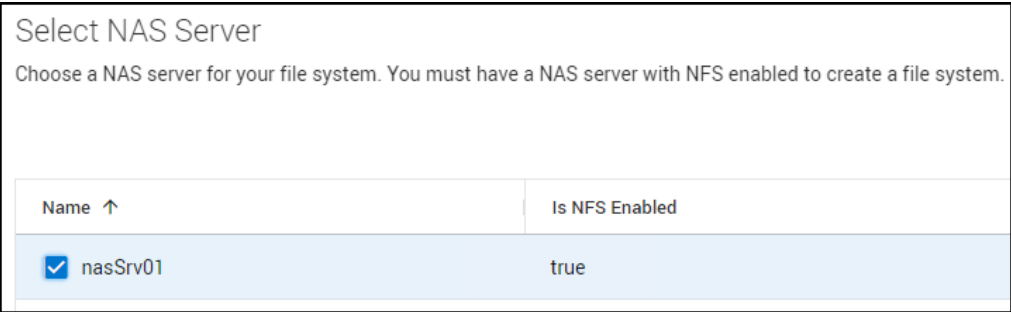
**Note:** A NAS Server must be created before a File System and NFS Export can be created. See the PowerStore documentation for the process to create a NAS Server.

To provision a File System and NFS Export in PowerStore Manager, follow these steps:

1. From the Storage | File Systems menu, click **Create**.



2. Select the NAS Server and click **Next**.



3. Specify a File System name and size and click **Next**.

### File System Details

Provide file system name and size

Name

nfs02

Description - optional

Size

200

GB

#### Advanced Settings

Host IO Size

8K

4. Specify an NFS Export name. Note the NFS Export details, which will be needed later. Click **Next**.



**NFS Export - optional**  
Provide NFS Export name

Name  
nfs02

Description - optional

NFS Export

NAS Server	nasSrv01
Local Path	/nfs02
File System Name	nfs02
NFS Export Path	100.88.145.146:/nfs02

5. On the Configure Access page, click **Add Host**, and add the VMkernel IP addresses of each vSphere host that will be used to mount the NFS volume. Each vSphere host requires an Access Type of **Read/Write, allow Root**. Click **Next**.

**Configure Access**

Minimum Security  
Sys

Default Access ⓘ  
No Access

+ Add Host Modify

☐ Name/Network Address

**Add Host**  
Add a group of hosts or select from a list

Access Type  
Read/Write, allow Root

ESXi Hosts/ IP Address ⓘ

ⓘ ESXi hosts require root access for NFSv3 and non-Kerberos NFSv4.1 protocols. ESXi host can have 'Read/Write' access for non-root Kerberos NFSv4 configurations. Refer to VMware NFS documentation for more information.

6. Apply an existing Protection Policy if wanted. Click **Next**.
7. Review the information on the summary page and click **Create File System**.
8. In the vSphere UI, follow the New Datastore workflow to mount the NFS Export.

New Datastore

1 Type

2 Select NFS version

3 Name and configuration

4 Ready to complete

Name and configuration

Specify datastore name and configuration.

If you plan to configure an existing datastore on new hosts in the datacenter, it is recommended to use the "Mount to additional hosts" action from the datastore instead.

NFS Share Details

Name: nfs02

Folder: nfs02

Server: /vols/vol0/datastore-001

Access Mode ☐ Mount NFS as read-only

CANCEL BACK NEXT

9. In the vSphere UI, right-click the new NFS datastore and mount the datastore to the remaining hosts in the vSphere cluster.

Mount Datastore to Additional Hosts | nfs02

COMPATIBLE (2 HOSTS) INCOMPATIBLE (0 HOSTS)

Filter

Host	Cluster
<input checked="" type="checkbox"/> s1250.techsol.local	<input checked="" type="checkbox"/> tanzu-cluster
<input checked="" type="checkbox"/> s1249.techsol.local	<input checked="" type="checkbox"/> tanzu-cluster

**Note:** Following best practices, be sure the NFS datastore is shared and mounted across all hosts in the vSphere cluster. If using NSX-T for networking and NFS datastores, VMware recommends a unique NFS datastore for each NSX Edge node.

## Virtual Volumes (vVols)

VMware Virtual Volumes is a storage framework which allows VM data to be stored on individual Virtual Volumes within a Storage Container. vVols allow data services to be applied at a VM-granularity level while using Storage Policy Based Management (SPBM). Using VM Storage Policies, vVol storage can be used for Supervisor Control Plane Nodes, Ephemeral Disks, Image Cache, and Persistent Volumes.

**Note:** At this point, a pattern has been established that using the vSphere CSI driver, PowerStore block volumes, NFS exports, and vVols are all suitable storage types for vSphere with Tanzu and Tanzu Kubernetes Clusters. It is up to the vSphere administrator to decide what type of storage to use for each component in vSphere with Tanzu based on requests made by the DevOps team. For more information about Dell Technologies CSI drivers, see <https://dell.github.io/csm-docs/docs/csdriver/>.

The PowerStore installation automatically provisions a default storage container for vVols. Before the storage container can be used, the PowerStore VASA provider must be registered with vCenter to present the storage container to the vSphere hosts. A vVol datastore is then created using the storage container. For more information about

registering the VASA provider and creating a vVol datastore, see the white paper [Dell PowerStore: Virtualization Integration](#).

After the PowerStore VASA provider has been registered with vCenter Server and the vVol datastore created, one or more VM Storage Policies must be established to allow vSphere with Tanzu and Tanzu Kubernetes Clusters to consume vVol storage.

---

**Note:** Like block volumes, a shares-based volume performance policy can be used with vVols to enforce the relative performance priority of a given vVol. Available options are Low, Medium (default), and High. The volume performance policy can be used as a storage tiering mechanism.

---

## Preparing and configuring vSphere and PowerStore

### Introduction

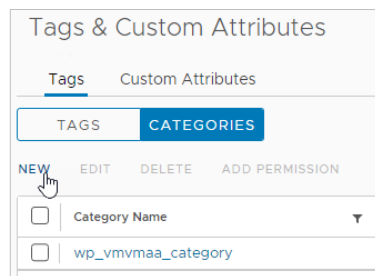
After PowerStore has been deployed, VM Storage Policies must be created and tied to the underlying storage. VM Storage Policies leverage VMFS block and file datastores by using tags. VM Storage Policies leverage vVol storage through the VASA provider. The following sections will provide guidance for configuring VM Storage Policies with tags and the VASA provider.

### Tags and categories

Tags allow vSphere administrators to attach metadata to objects in the vSphere inventory to make it easier to sort and search for these objects. Tags can also be used with VM Storage Policies for Storage Policy Based Management (SPBM). vSphere with Tanzu and Tanzu Kubernetes Clusters use VM Storage Policies for placement of infrastructure objects and to define storage classes for persistent volumes.

To create a tag category, follow these steps:

1. In the vSphere UI, go to **Tags & Custom Attributes**.
2. Create a category.



3. Assign a **Category Name**. For **Associable Object Types**, select **Datastore**; optionally, also select **Datastore Cluster** if datastore clusters will be used. Click **Create**.

Create Category

Category Name:

vk8s

Description:

Tags Per Object:

☒ One tag

☐ Many tags

Associable Object

☐ All objects

Types:

☐ Folder

☐ Datacenter

☒ Datastore Cluster

☐ Distributed Switch

☐ Content Library

☐ Network

☐ vApp

☐ Cluster

☐ Datastore

☐ Distributed Port Group

☐ Host

☐ Library Item

☐ Resource Pool

☐ Virtual Machine

CANCEL

CREATE

**Note:** VMware recommends using Datastore Clusters and Storage DRS for Supervisor Cluster Control Plane Nodes. However, vSphere Pods enable SCSI bus sharing and if they are deployed on a Datastore Cluster, the individual datastores in the Datastore Cluster cannot be placed in maintenance mode. A Datastore Cluster loses some of its functionality if used for placement of vSphere with Tanzu Ephemeral Disks.

To create a tag, follow these steps:

1. In the vSphere UI, go to **Tags & Custom Attributes**.
2. Create a tag.

Tags & Custom Attributes

Tags

Custom Attributes

TAGS

CATEGORIES

NEW

EDIT

DELETE

ADD PERMISSION

☐

Tag Name

☐

wp\_vmvmaa\_tag

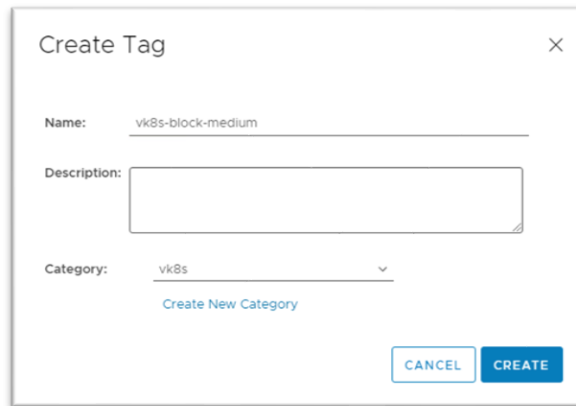
☐

wp\_tkg\_ctrl\_vmvmaa\_tag

☐

wp\_tkg\_worker\_vmvmaa\_tag

3. Assign a Name that describes the type and tier of storage the tag will be applied to. Select the tag category created in a previous step. Click **Create**.



**Create Tag**

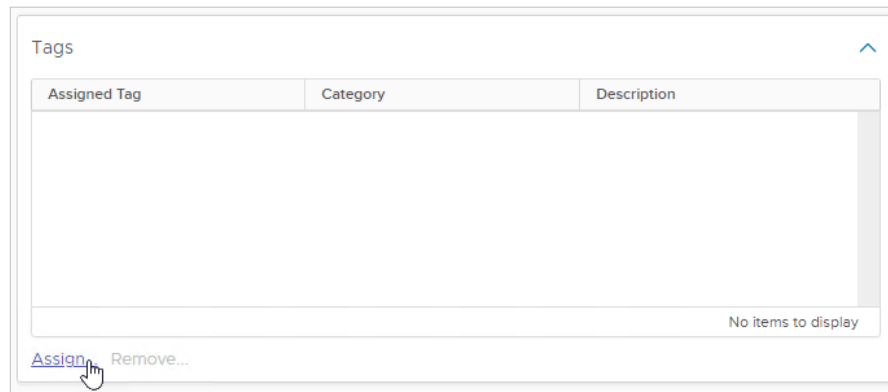
Name:

Description:

Category:  [Create New Category](#)

To apply the tag to a PowerStore backed VMFS or NFS datastore, follow these steps:

1. In the vSphere UI, go to Storage.
2. Identify the datastore to be tagged and left click on it in the inventory tree.
3. Under **Tags**, click **Assign**.

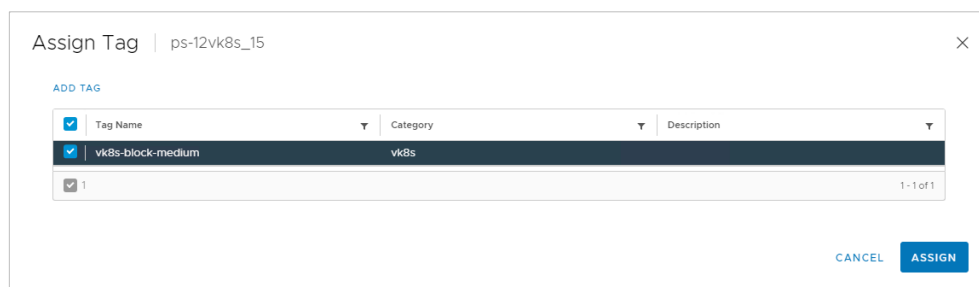


Tags

Assigned Tag	Category	Description
No items to display		

[Assign](#) [Remove...](#)

4. Choose the tag created in an earlier step that describes the type and tier of storage. Click **Assign**.



Assign Tag | ps-12vk8s\_15

ADD TAG

<input checked="" type="checkbox"/>	Tag Name	Category	Description
<input checked="" type="checkbox"/>	vk8s-block-medium	vk8s	

☒ 1 1 - 1 of 1

## VM Storage Policies

VM Storage Policies are typically associated with traditional virtual machine workloads. VM Storage Policies are used to tie application storage requirements to underlying block, NFS, or vVol storage capabilities advertised through tags or the VASA provider respectively. In the context of vSphere with Tanzu, VM Storage Policies are used for the same purpose – to tie workload requirements to capable underlying storage. However, in the case of vSphere with Tanzu, workloads are not traditional VMs. Instead, they are Supervisor Control Plane Nodes, Ephemeral Disks, Image Cache, and Persistent Volumes.

In addition, VM Storage Policies are closely related to Kubernetes storage classes. When a VM Storage Policy is created and assigned for use in vSphere with Tanzu, vSphere with Tanzu creates a matching Kubernetes storage class in the Supervisor Namespace. These storage classes will also be replicated to any VMware Tanzu Kubernetes Grid Cluster that is deployed.

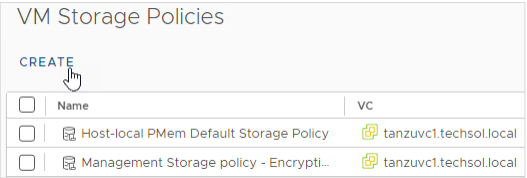
Creating a VM Storage Policy is the final configuration needed before vSphere with Tanzu can be deployed and examples will be provided in the following sections.

**Note:** VMware is flexible with the characters allowed in VM Storage Policy names. However, storage class names are restrictive. For example, spaces and upper-case characters are not allowed in storage class names. Since VM Storage Policies names become storage class names in vSphere with Tanzu and Tanzu Kubernetes Grid Clusters, illegal characters used in a VM Storage Policy name must be resolved. VMware handles this automatically by replacing spaces with the '-' character and converting upper-case characters to lower-case characters. As a best practice, do not use spaces and upper-case characters in VM Storage Policy names. Use a naming convention compliant with storage class names.

### Block and NFS storage

To create a VM Storage Policy for block or NFS storage on PowerStore, follow these steps:

1. In the vSphere UI, go to Policies & Profiles | VM Storage Policies.
2. Click **Create**.



3. Assign a Name that describes the type and tier of storage. Click **Next**.

4. Choose **Enable tag based placement rules**. Click **Next**.

**Create VM Storage Policy**

- 1 Name and description
- 2 Policy structure**
- 3 Tag based placement
- 4 Storage compatibility
- 5 Review and finish

**Policy structure**

Host based services

Create rules for data services provided by hosts. Available data services could include encryption, I/O control, caching, etc. Host based services will be applied in addition to any datastore specific rules.

☐ Enable host based rules

Datastore specific rules

Create rules for a specific storage type to configure data services provided by the datastores. The rules will be applied when VMs are placed on the specific storage type.

☐ Enable rules for "vSAN" storage

☐ Enable rules for "vSANDirect" storage

☐ Enable rules for "DELLPOWERSTORE.VVOL" storage

☒ Enable tag based placement rules

CANCEL BACK NEXT

- Choose the Tag category created earlier. For **Usage option**, select **Use storage tagged with**. Click **Browse Tags** and select an appropriate tag created earlier. In this example, a silver policy and storage class are being created, which tie to a block volume having a medium volume performance policy. Click **Next**.

**Create VM Storage Policy**

- 1 Name and description
- 2 Policy structure
- 3 Tag based placement**
- 4 Storage compatibility
- 5 Review and finish

**Tag based placement**

Add tag rules to filter datastores to be used for placement of VMs.

Rule 1 REMOVE

Tag category vk8s

Usage option Use storage tagged with

Tags vk8s-block-me... X

BROWSE TAGS

ADD TAG RULE

CANCEL BACK NEXT

- Verify the expected storage compatibility. In this example, it is a block volume provisioned in an earlier step. Click **Next**.

Create VM Storage Policy

1 Name and description

2 Policy structure

3 Tag based placement

4 Storage compatibility

5 Review and finish

Storage compatibility

COMPATIBLE

INCOMPATIBLE

☐ Expand datastore clusters

Compatible storage 199.75 GB (198.34 GB free)

Filter

Name	Datacenter	Type	Free Space	Capacity	Warnings
ps-12vk8s_15	Tanzu Datacenter	VMFS 6	198.34 GB	199.75 GB	

1 item

CANCEL

BACK

NEXT

7. Review the new VM Storage Policy and Tag association. Click **Finish**.

Create VM Storage Policy

1 Name and description

2 Policy structure

3 Tag based placement

4 Storage compatibility

5 Review and finish

Review and finish

General

Name

vk8s-block-silver

Description

vCenter Server

tanzuvcl.techsol.local

Tag based placement

Tagged with: vk8s

vk8s-block-medium

CANCEL

BACK

FINISH

vVol storage

To create a VM Storage Policy for vVol storage on PowerStore, follow these steps:

- 1. In the vSphere UI, go to Policies & Profiles | VM Storage Policies.
- 2. Click **Create**.

VM Storage Policies

CREATE

<input type="checkbox"/>	Name	VC
<input type="checkbox"/>	Host-local PMem Default Storage Policy	tanzuvcl.techsol.local
<input type="checkbox"/>	Management Storage policy - Encrypti...	tanzuvcl.techsol.local

- 3. Assign a Name that describes the type and tier of storage. Click **Next**.



### Create VM Storage Policy

- 1 Name and description**
- 2 Policy structure
- 3 Dell EMC PowerStore rules
- 4 Storage compatibility
- 5 Review and finish

#### Name and description

vCenter Server: TANZUVCI.TECHSOL.LOCAL

Name:

Description:

CANCEL NEXT

4. Select **Enable rules for “DELLEMC.POWERSTORE.VVOL” storage**. Click **Next**.

### Create VM Storage Policy

- 1 Name and description
- 2 Policy structure**
- 3 Dell EMC PowerStore rules
- 4 Storage compatibility
- 5 Review and finish

#### Policy structure

Host based services

Create rules for data services provided by hosts. Available data services could include encryption, I/O control, caching, etc. Host based services will be applied in addition to any datastore specific rules.

☐ Enable host based rules

Datastore specific rules

Create rules for a specific storage type to configure data services provided by the datastores. The rules will be applied when VMs are placed on the specific storage type.

☐ Enable rules for "vSAN" storage

☐ Enable rules for "vSANDirect" storage

☐ Enable rules for "VMFS" storage

☒ Enable rules for "Dell EMC PowerStore" storage

☐ Enable tag based placement rules

Storage topology

Create rules for storage consumption domain topology. The storage topology will be applied to all datastore specific rules.

☐ Enable consumption domain

CANCEL BACK NEXT

5. Select a QoS Priority. This is the shares-based volume performance policy. Click **Next**.

### Create VM Storage Policy

- 1 Name and description
- 2 Policy structure
- 3 Dell EMC PowerStore rules**
- 4 Storage compatibility
- 5 Review and finish

#### Dell EMC PowerStore rules

Placement Tags

QoS Priority ⓘ MEDIUM

CANCEL BACK NEXT

6. Verify the expected storage compatibility. In this example, it is the default storage container vVol datastore. Click **Next**.

Create VM Storage Policy

1 Name and description

2 Policy structure

3 Dell EMC PowerStore rules

4 Storage compatibility

5 Review and finish

Storage compatibility

COMPATIBLE

INCOMPATIBLE

☐ Expand datastore clusters

Compatible storage 58.23 TB (58.23 TB free)

Quick Filter

Enter value

Name	Datacenter	Type	Free Space	Capacity	Warnings
PowerStore PS-3	Datacenter	vVol	58.23 TB	58.23 TB	

Manage Columns

1 item

CANCEL

BACK

NEXT

7. Review the new VM Storage Policy and associated vVol rules. Click **Finish**.

Create VM Storage Policy

1 Name and description

2 Policy structure

3 Dell EMC PowerStore rules

4 Storage compatibility

5 Review and finish

Review and finish

General

Name

vk8s-vvol-silver

Description

vCenter Server

tanzuvcl1.techsol.local

Dell EMC PowerStore rules

Placement

QoS Priority

MEDIUM

CANCEL

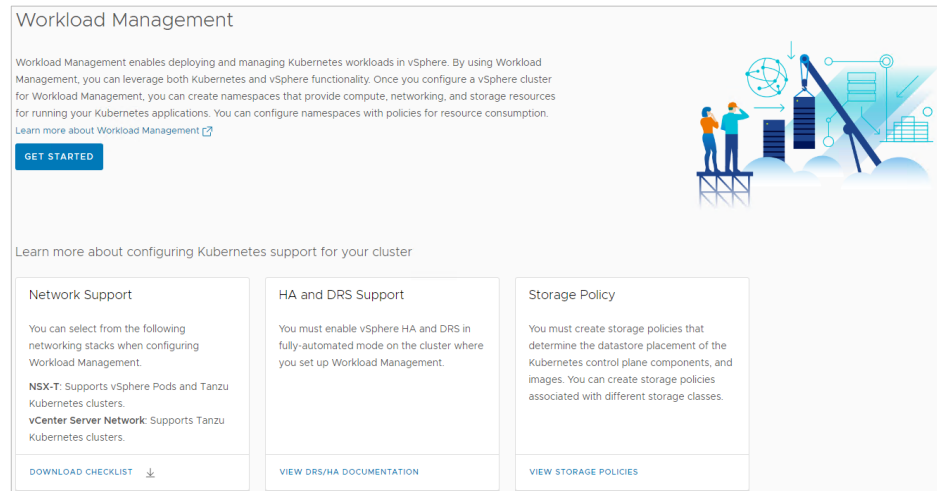
BACK

FINISH

## Deploying vSphere with Tanzu

After PowerStore storage and vSphere infrastructure is prepared and configured, vSphere with Tanzu can be deployed. Deployment is accomplished by enabling Workload Management in the vSphere UI. VMware details this process in the document [vSphere with Tanzu Configuration and Management](#). The following steps are shown as an example using NSX-T for networking and PowerStore for the storage:

1. In the vSphere UI, use the pulldown menu to go to **Workload Management**, and then click **Get Started**.



**Figure 4. Enabling Workload Management to deploy vSphere with Tanzu**

2. Select **NSX-T** to support vSphere Pods and Tanzu Kubernetes clusters.

1. vCenter Server and Network
Select a vCenter Server and a network to enable a cluster

To enable Workload Management on a cluster, select the vCenter Server system that hosts the cluster.

**Select a vCenter** TANZUVCI.TECHSOL.LOCAL (SUPPORTS NSX-T) ⓘ

Select the networking stack that will provide connectivity to the Workload Management platform.

**Select a networking stack option**

☒ **NSX-T**  
Supports vSphere Pods and Tanzu Kubernetes clusters.

☐ **vCenter Server Network**  
Supports Tanzu Kubernetes clusters.

**NEXT**

3. Select the DRS and HA cluster where vSphere with Tanzu will be deployed.

2. Select a Cluster Select a cluster to enable Workload Management

Select a cluster to enable Workload Management. This cluster must have enough space to run your Kubernetes workloads and three Kubernetes control plane VMs.

>

tanzuvc1.techsol.local

Cluster Details | tanzuvc1.techsol.local

COMPATIBLE

INCOMPATIBLE ⓘ

Cluster Name	Number of Hosts	Available CPU	Available Memory
tanzu-cluster	3	107.12 GHz	302.5 GB

1 item

NEXT

#### 4. Select a Control Plane Size.

3. Control Plane Size Select the size and resources available for control plane VM on this cluster

Allocate capacity for the Kubernetes control plane VMs. The amount of resources that you allocate to the control plane VMs determines the amount of Kubernetes workloads the cluster can support.

Resource allocation

☐ Tiny - 2 CPUs, 8 GB Memory, 16 GB Storage  
☒ Small - 4 CPUs, 16 GB Memory, 16 GB Storage  
☐ Medium - 8 CPUs, 24 GB Memory, 16 GB Storage  
☐ Large - 16 CPUs, 32 GB Memory, 16 GB Storage

NEXT

#### 5. Select the VM Storage Policies to be used for the Control Plane Nodes, Ephemeral Disks, and Image Cache. The VM Storage Policies were created earlier and will dictate which storage each of the vSphere with Tanzu components lands on. Each designation can be modified later. However, a modification does not trigger an immediate storage migration. Consult the vSphere with Tanzu documentation for more information about this process.

4. Storage Select the storage policy to the Control Plane VMs

Select a storage policy to be used for datastore placement of Kubernetes control plane components. The policy is associated with a datastore on the vSphere environment.

Control Plane Nodes	vk8s-block-silver	VIEW DATASTORES
Ephemeral Disks	vk8s-vvol-silver	VIEW DATASTORES
Image Cache	vk8s-nfs-silver	VIEW DATASTORES

NEXT

VM Storage Policies

**Note:** VMware recommends using Storage DRS with anti-affinity rules to maintain Control Plane Node VMs on separate datastores. The datastore Tag and VM Storage Policy must apply to all the datastores in the datastore cluster, and the datastore cluster itself. Deploy the Control Plane Nodes to the VM Storage Policy associated with the datastore cluster. After deployment, create and apply the Storage DRS anti-affinity rules which will move the Control Plane Nodes so that each is on a unique datastore.

vSphere Pods enable SCSI bus sharing and if they are deployed on a Datastore Cluster, the individual datastores in the Datastore Cluster cannot be placed in maintenance mode. A Datastore Cluster loses some of its functionality if used for placement of vSphere with Tanzu Ephemeral Disks.

## 6. Provide details for the Management Network.

5. Management Network Configure Management network for the Control Plane and Worker nodes

The Workload Management consists of three Kubernetes control plane VMs and the Spherelet process on each host, which allows the hosts to be joined in a Kubernetes cluster. The cluster where you set up Workload Management is connected to a management network supporting traffic to vCenter Server.

[VIEW NETWORK TOPOLOGY](#)

Network 1

Starting IP Address 1

Subnet Mask 1

Gateway 1

DNS Server 1

DNS Search Domains (Optional) 1

NTP Server 1

[NEXT](#)

## 7. Provide details for the Workload Network.

6. Workload Network Configure networking to support traffic to the Kubernetes API and to workloads and services.

The workload network supports traffic to the Kubernetes API and to the Pods/Services that are deployed on the Supervisor Cluster. This network is supported by NSX.

vSphere Distributed Switch 1  Edge Cluster 1

API Server endpoint FQDN (Optional) 1

DNS Servers 1

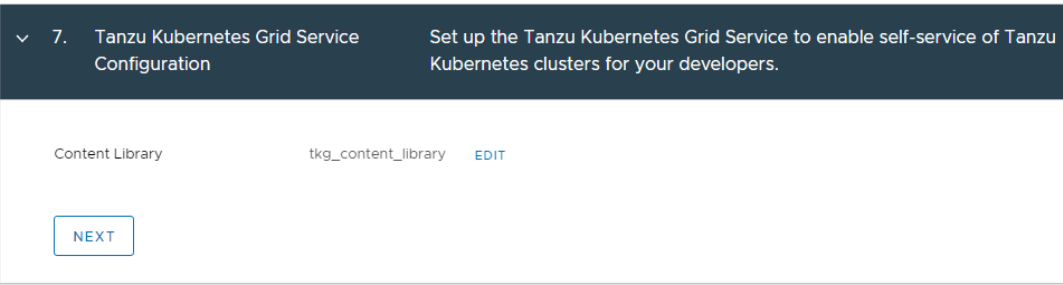
Pod CIDRs 1  Service CIDR 1

This field cannot be edited later once saved. Make sure all CIDR values are unique.

Ingress CIDRs 1  Egress CIDRs 1

[NEXT](#)

## 8. Select the Content Library to be used for the Tanzu Kubernetes Grid Service. At the time of this writing, the Content Library Subscription URL is <https://wp-content.vmware.com/v2/latest/lib.json>.



9. Review and Confirm. The Supervisor Control Plane Nodes will be deployed. Installation typically takes 30 to 60 minutes depending on the environment.

Once the deployment is complete, use the vSphere UI to go to Hosts and Clusters. Left click on the cluster object. From Monitor | Namespaces | Overview page provides information about the Supervisor Cluster, Kubernetes version and status, and the Control Plane Nodes. The Control Plane Node IP Address should be HTTPS accessible in a web browser from the workload/DevOps network. This is where the Kubernetes CLI Tools can be downloaded and installed from. After the Kubernetes CLI Tools are installed, the Control Plane Node IP Address serves as the API endpoint that kubectl commands will be issued against.

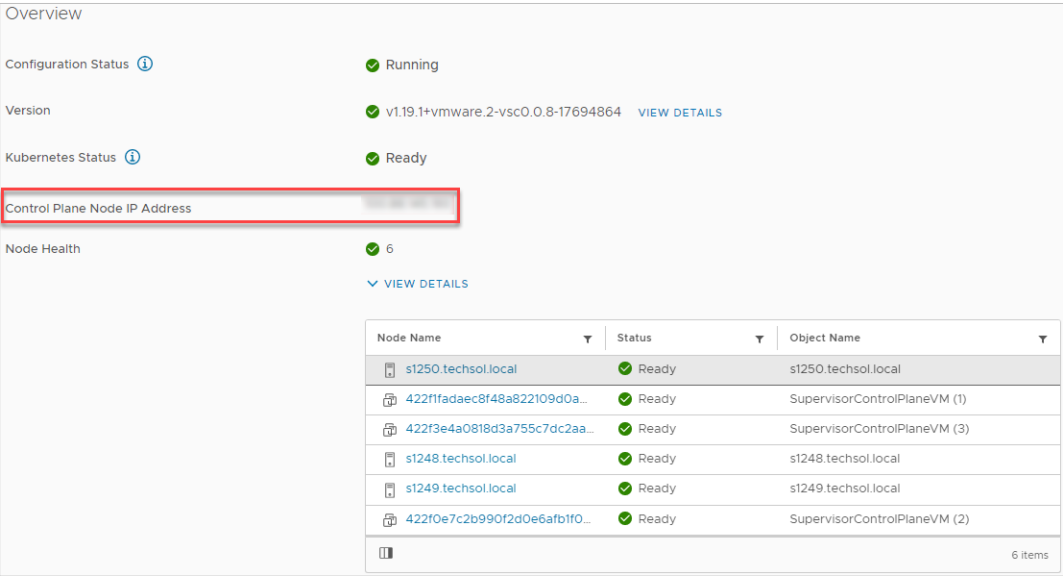


Figure 5. vSphere with Tanzu Supervisor Cluster Overview and Control Plane Node IP Address

## Day 2 Kubernetes

### Introduction

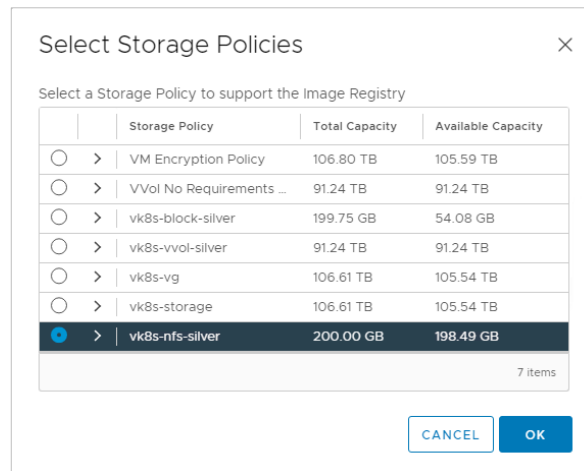
With design and deployment (sometimes referred to as Day 0 and Day 1) complete, we can now start to focus on some of the operational aspects of vSphere with Tanzu and Tanzu Kubernetes Clusters along with their relationship to PowerStore storage. The following sections will explore these areas.

## Enabling the Harbor Image Registry

vSphere with Tanzu ships with Harbor, which can be used as a local or private DevOps image registry. For some vSphere administrators, Harbor may be the first application deployed in the vSphere with Tanzu environment at the request of the DevOps team. Before Harbor can be used, it must be enabled.

To enable the Harbor, follow these steps:

1. In the vSphere UI, go to Hosts and Clusters. Left-click the cluster object. From Configure | Namespaces | Image Registry click **Enable Harbor**.
2. Select the VM Storage Policy that will be used to store the images. In this example, **vk8s-nfs-silver** is used.



The deployment process takes a few minutes, and its progress can be observed in the vSphere UI as the vSphere Pods are instantiated on the Supervisor Cluster. Once the deployment is complete, verify the Harbor health and storage designation in the vSphere UI. The Harbor UI address is shown and should be HTTPS accessible in a web browser.

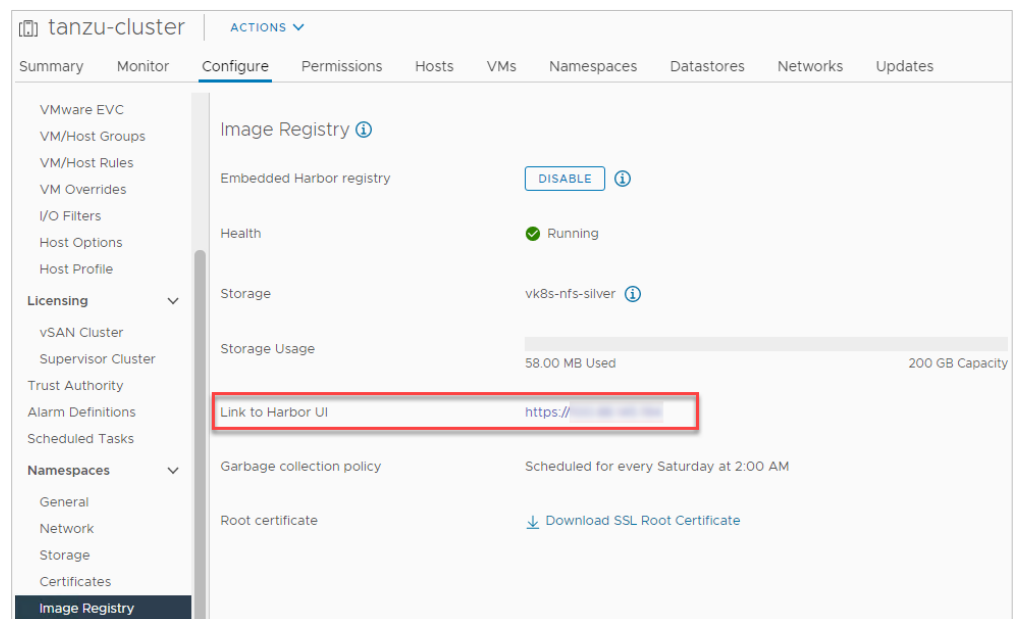
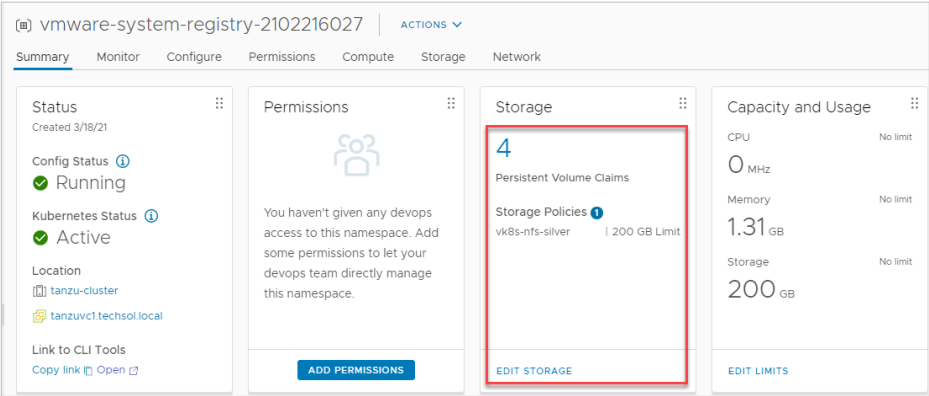


Figure 6. The Harbor Image Registry status page

Use the vSphere UI to examine the namespace where Harbor is deployed. In this example, the namespace is **vmware-system-registry-2102216027**. There are a few items to highlight in the Storage section.

- The Namespace is assigned the **vk8s-nfs-silver** Storage Policy. This policy ties to the NFS Export **nfs02** created previously.
- There is a 200 GB storage limit applied to the **vk8s-nfs-silver** Storage Policy in the Namespace.
- The Storage Policy in use can be modified using the **Edit Storage** hyperlink.
- There are four Persistent Volume Claims in use.

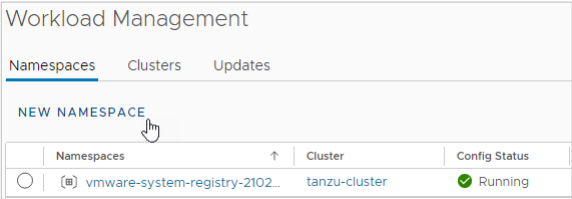


**Figure 7. Summary page of the Harbor application's Namespace**

### Creating a vSphere Namespace

Containerized applications are deployed across one or more vSphere Pods in a vSphere Namespace. A vSphere Namespace is backed by a Resource Pool which is used to control resource limits. In the following example, a new vSphere Namespace will be created to explore storage integration further. To create a vSphere Namespace, follow these steps:

1. In the vSphere UI, go to Workload Management. Click **New Namespace**.



2. Provide a DNS-compliant name. This example uses the name **devops**. Click **Create**.



---

**Note:** vSphere Namespace names must be DNS-compliant. The character requirements are:  
 Alphanumeric (a-z and 0-9) string with maximum length of 63 characters.  
 '-' character allowed anywhere except the first or last character.  
 Must be unique across all namespace enabled clusters within a vCenter Server.

---

The devops vSphere Namespace is now created. The next step would typically be to delegate permissions to a DevOps team. These steps are covered in depth in the VMware document [vSphere with Tanzu Configuration and Management](#).

## Deploy an application in a vSphere Pod

With the **devops** vSphere Namespace created, Kubernetes CLI Tools can be used to deploy a containerized application. The following example is a simple demonstration application named [hello-kubernetes](#) which deploys a LoadBalancer and three vSphere Pod replicas.

```
C:\>kubectl vsphere login --server=https://xxx.xxx.xxx.193 --vsphere-username administrator@vsphere.local --insecure-skip-tls-verify
```

```
Password:
Logged in successfully.
```

```
You have access to the following contexts:
xxx.xxx.xxx.193
devops
```

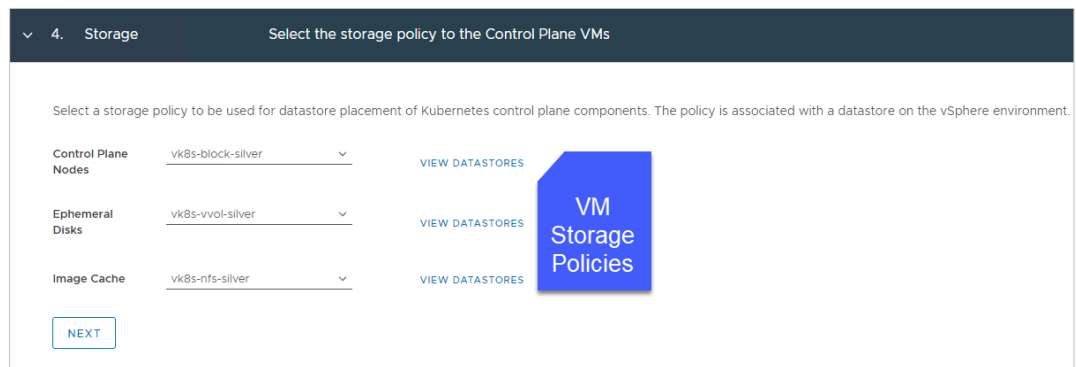
If the context you wish to use is not in this list, you may need to try logging in again later, or contact your cluster administrator.

To change context, use ``kubectl config use-context <workload name>``

```
C:\>kubectl config use-context devops
Switched to context "devops".
```

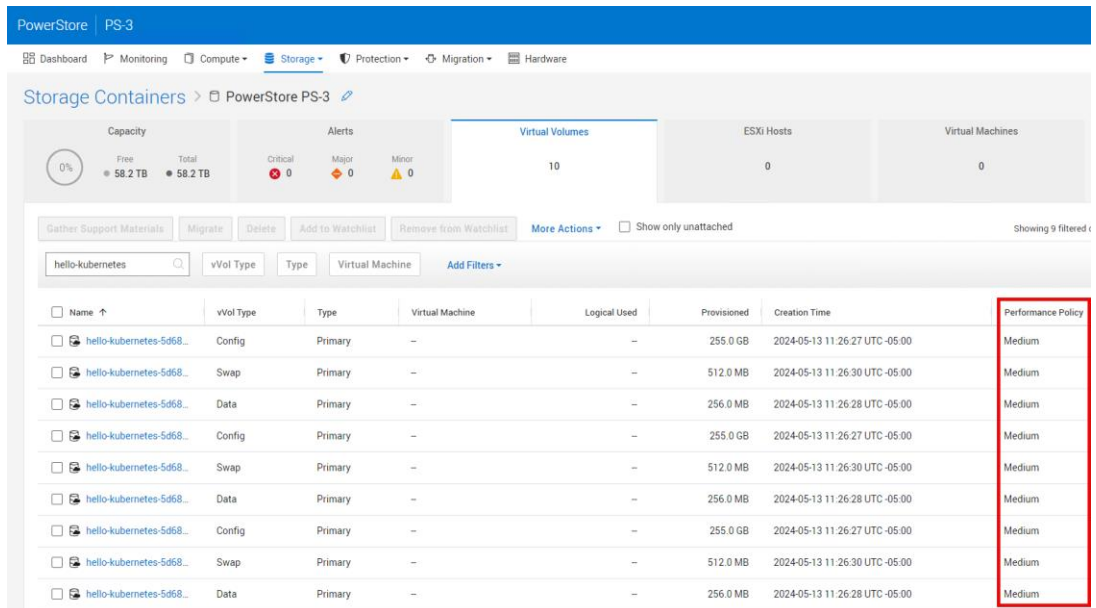
```
C:\>kubectl apply -f C:\Users\jboche\Downloads\Kubernetes-
master\Kubernetes-master\demo-applications\demo-
hellokubernetes.yaml
service/hello-kubernetes created
deployment.apps/hello-kubernetes created
```

Recalling the vSphere with Tanzu deployment, the **vk8s-vvol-silver** VM Storage Policy was specified for Ephemeral Disks. Ephemeral Virtual Disks are transient storage for vSphere Pods which exist throughout the life cycle of the vSphere Pod.



**Figure 8.** The **vk8s-vvol-silver** VM Storage Policy specified for Ephemeral Disks

Each vSphere Pod has one ephemeral disk that exists throughout the life cycle of the vSphere Pod. Since the **hello-kubernetes** application deployed three vSphere Pods, three sets of vVols will be created for the **hello-kubernetes** application. These vVols are not only visible in the vSphere UI, but they are also visible in the PowerStore Manager figure shown below. A filter is enabled to show only the Config and Data vVols for the hello-kubernetes application.



**Figure 9. Examining vSphere Pod consumption of vVol storage**

The IO Priority column is another area of interest. When the **vk8s-vvol-silver** VM Storage Policy was created, a QoS Priority of Medium was chosen. Looking at the figure above, it is clear that the **hello-kubernetes** application is tied to a medium IO Priority. If there is significant storage IO contention, the **hello-kubernetes** application will have priority over Low IO Priority workloads, but would yield to High Priority workloads. This is a working example of how a volume performance policy can be used as a storage tiering mechanism for Kubernetes workloads.

The last item to note is that storage has not yet been added to the **devops** vSphere Namespace. Storage is added to a vSphere Namespace to support persistent volumes and persistent volume claims. I know that the **hello-kubernetes** application does not use persistent storage so there was not a need to add storage to this vSphere Namespace yet. Adding storage will be covered in an upcoming section.

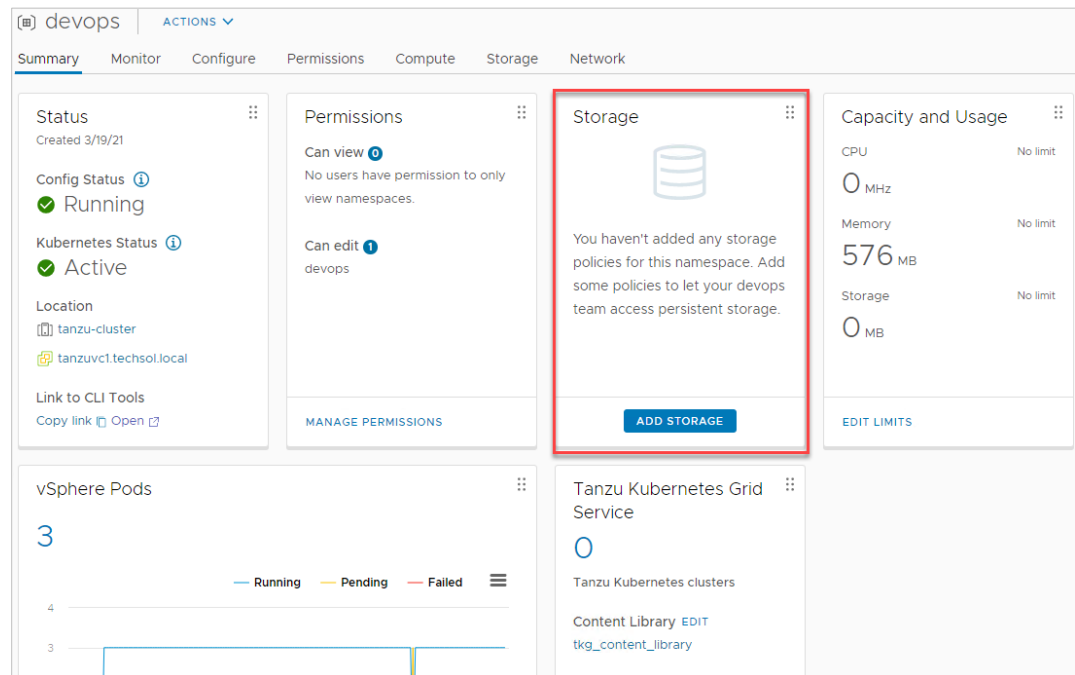


Figure 10. Devops vSphere Namespace showing no storage assignment

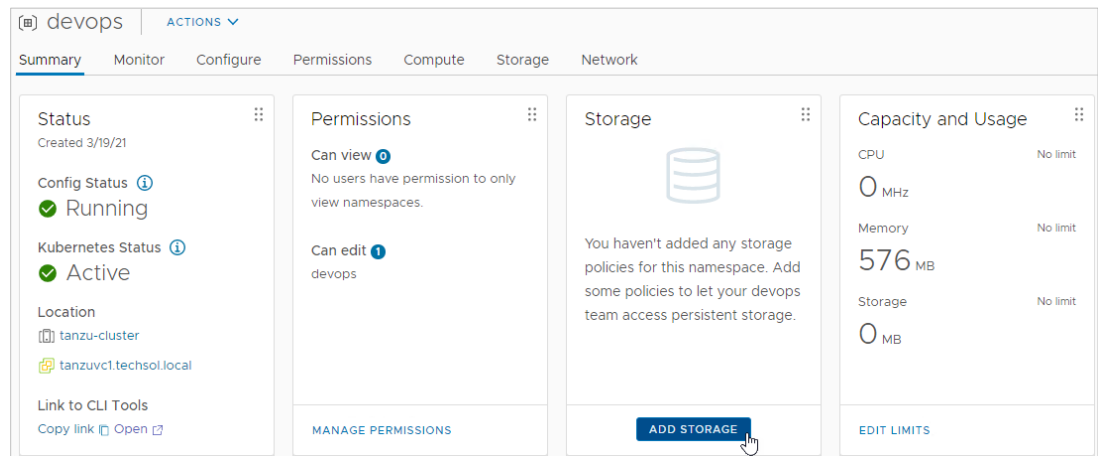
### Adding, limiting, and removing storage in a vSphere Namespace

Some applications do not maintain stateful, or persistent data therefore they have no need for persistent volumes in vSphere with Tanzu. This was demonstrated with the **hello-kubernetes** application in a previous section. However, most applications maintain stateful data of some type and will need persistent storage in vSphere with Tanzu. This section will outline the steps for adding storage in a vSphere Namespace and setting limits on storage consumption.

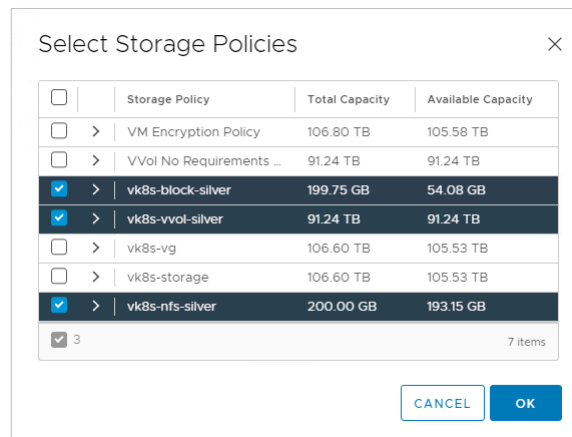
#### Adding storage in a vSphere Namespace

Before a DevOps team can request persistent storage through a persistent volume claim, storage that meets the requirements needs to be added to the vSphere Namespace. To add storage to a vSphere Namespace, follow these steps:

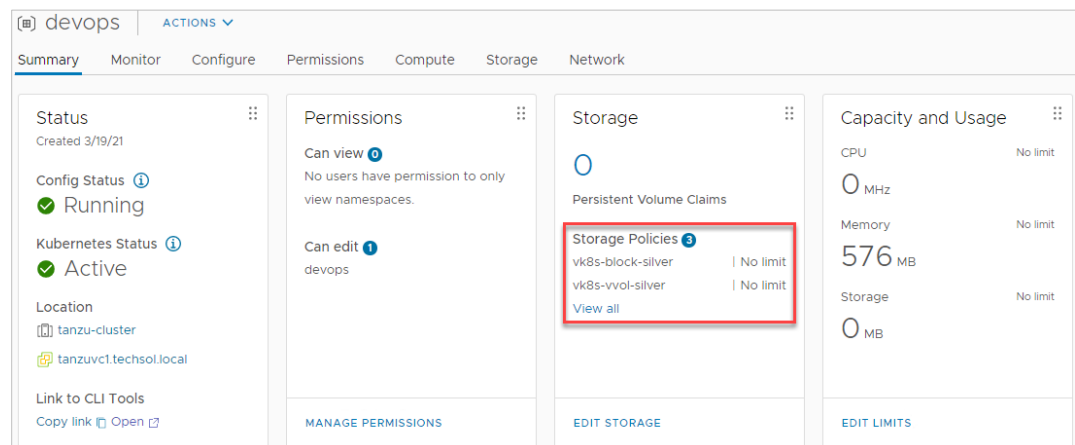
1. In the vSphere UI, go to Workload Management. Go to the vSphere Namespace requiring persistent storage. Click **Add Storage**.



2. Select one or more Storage Policies to be used to back persistent storage. When a VM Storage Policy is created and assigned for use in vSphere with Tanzu, vSphere with Tanzu creates a matching Kubernetes storage class in the Supervisor Namespace. These storage classes will also be replicated to any VMware Tanzu Kubernetes Grid Cluster that is deployed. The storage class will be used in the persistent volume claim for persistent storage. After the selection is made, click **OK**.



After following the previous steps, the Storage Policies should be visible in the vSphere UI.



**Figure 11. Storage Policies available for use as persistent storage in the vSphere Namespace**

A logged-in DevOps user is also able to see the same Storage Policies available as storage classes using the Kubernetes CLI Tools.

```
C:\>kubectl vsphere login --server=https://xxx.xxx.xxx.193 --
vsphere-username devops@vsphere.local --insecure-skip-tls-verify
```

Password:

Logged in successfully.

You have access to the following contexts:

```
xxx.xxx.xxx.193
devops
```

If the context you wish to use is not in this list, you may need to try logging in again later, or contact your cluster administrator.

To change context, use ``kubectl config use-context <workload name>``

```
C:\>kubectl config use-context devops
Switched to context "devops".
```

```
C:\>kubectl get storageclass
```

NAME	PROVISIONER	RECLAIMPOLICY
VOLUMEBINDINGMODE	ALLOWVOLUMEEXPANSION	AGE
vk8s-block-silver	csi.vsphere.vmware.com	Delete
Immediate	true	11m
vk8s-nfs-silver	csi.vsphere.vmware.com	Delete
Immediate	true	21h
vk8s-vvol-silver	csi.vsphere.vmware.com	Delete
Immediate	true	11m

Using the **kubectl describe storageclass** command, more information is displayed about each storage class including the CSI driver. Note in the following example and throughout this document, the vSphere CSI driver is used.

```
C:\>kubectl describe storageclass
```

```
Name: vk8s-block-silver
IsDefaultClass: No
Annotations:
  cns.vmware.com/StoragePoolTypeHint=cns.vmware.com/VMFS
Provisioner: csi.vsphere.vmware.com
Parameters: storagePolicyID=475fdecf-f339-4f99-9013-87edd1ab6cdb
AllowVolumeExpansion: True
MountOptions: <none>
ReclaimPolicy: Delete
VolumeBindingMode: Immediate
Events: <none>
```

```

Name: vk8s-nfs-silver
IsDefaultClass: No
Annotations:
  cns.vmware.com/StoragePoolTypeHint=cns.vmware.com/NFS
Provisioner: csi.vsphere.vmware.com
Parameters: storagePolicyID=716b2ccb-ea39-42e9-8321-545b47f1102e
AllowVolumeExpansion: True
MountOptions: <none>
ReclaimPolicy: Delete
VolumeBindingMode: Immediate
Events: <none>

```

```

Name: vk8s-vvol-silver
IsDefaultClass: No
Annotations:
  cns.vmware.com/StoragePoolTypeHint=cns.vmware.com/VVOL
Provisioner: csi.vsphere.vmware.com
Parameters: storagePolicyID=85e93f9f-ed1c-43ad-9263-0b6395e39b75
AllowVolumeExpansion: True
MountOptions: <none>
ReclaimPolicy: Delete
VolumeBindingMode: Immediate
Events: <none>

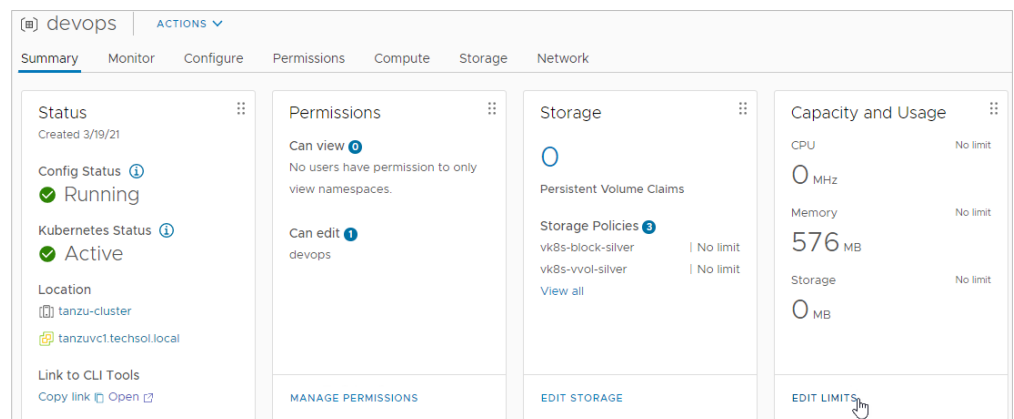
```

## Limiting storage in a vSphere Namespace

A vSphere Namespace is backed by a resource pool which has the ability to control resource consumption within the namespace. Typical resource limiting might be applied to CPU and memory consumption across all the vSphere Pods in the vSphere Namespace. However, limits can be applied to persistent storage as well.

To add global storage limit to a vSphere Namespace, follow these steps:

1. In the vSphere UI, go to Workload Management. Go to the vSphere Namespace. Under Capacity and Usage, click **Edit Limits**.



2. Provide the storage limit in terms of MB or GB and click **OK**.

**Resource Limits** [X]

Below are various resources that are available to the namespace. You can choose to limit consumption of any or all of these. This is an optional step.

CPU	No limit	MHz
Memory	No limit	MB
> Storage	50	GB

[CANCEL](#) [OK](#)

After following the previous steps, the storage limit should be visible in the vSphere UI.

The screenshot shows the vSphere UI for the 'devops' namespace. The 'Summary' tab is active, and the 'Storage' section is expanded. In the 'Capacity and Usage' panel, the 'Storage' row shows a '50 GB Limit', which is highlighted with a red rectangle.

**Figure 12. Viewing a global storage limit in a vSphere Namespace**

Using the **kubectl describe namespace** command, the same 50 GB storage limit is visible as a 50Gi hard storage quota.

```
C:\>kubectl describe namespace devops
Name:          devops
Labels:        vSphereClusterID=domain-c8
Annotations:   ls_id-0: cea4a565-dfc7-4ee2-a79c-137f9723a86c
               ncp/extpoolid: domain-c8:c2fdf804-b0a0-4bcb-99be-
9dab04afa64f-ippool-100-88-145-225-100-88-145-254
               ncp/router_id: t1_556a87e1-2308-4992-8418-
1445bd1b4dd5_rtr
               ncp/snat_ip: xxx.xxx.xxx.227
               ncp/subnet-0: 10.244.0.32/28
               vmware-system-resource-pool: resgroup-3033
               vmware-system-resource-pool-cpu-limit:
               vmware-system-resource-pool-memory-limit:
               vmware-system-vm-folder: group-v3034
Status:        Active

Resource Quotas
Name:          devops
Resource      Used  Hard
-----
requests.storage  0   50Gi
```



Name:  
devops-storagequota

Resource

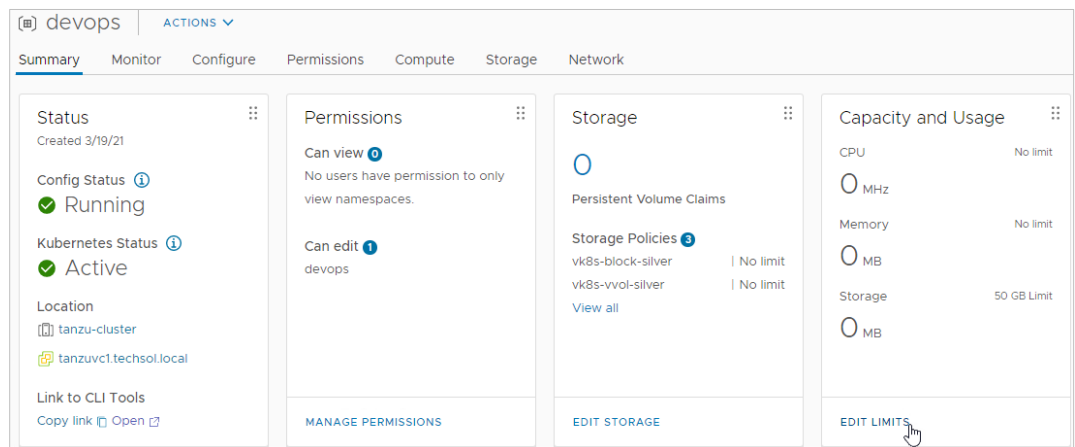
Used Hard

-----	---	--
vk8s-block-silver.storageclass.storage.k8s.io/requests.storage	0	
9223372036854775807		
vk8s-nfs-silver.storageclass.storage.k8s.io/requests.storage	0	
9223372036854775807		
vk8s-vvol-silver.storageclass.storage.k8s.io/requests.storage	0	
9223372036854775807		

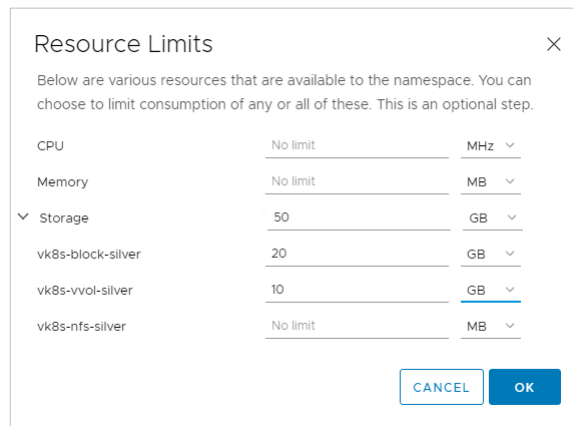
No LimitRange resource.

To add individual per storage limits to a vSphere Namespace, follow these steps:

1. In the vSphere UI, go to Workload Management go go into to the vSphere Namespace. Under Capacity and Usage, click **Edit Limits**.



2. Expand the Storage category and provide the storage limits for each storage policy in terms of MB or GB. Click **OK**.



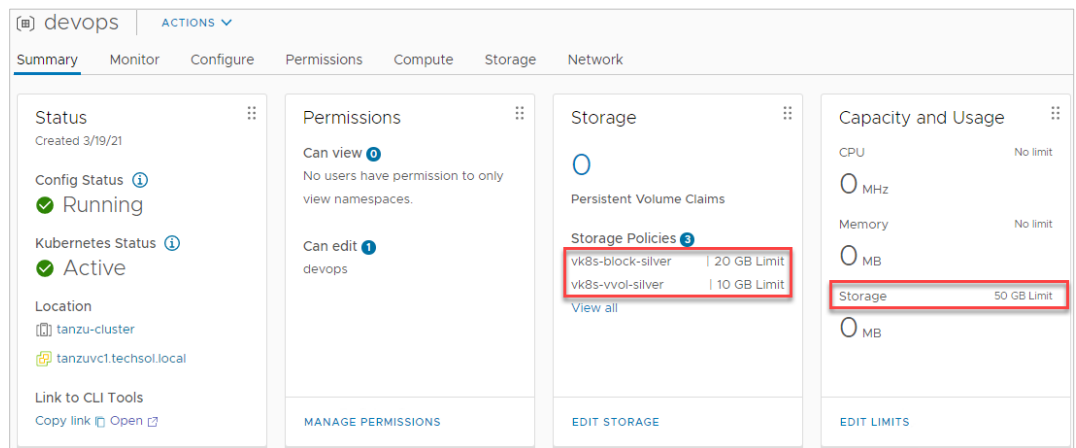
**Resource Limits** [X]

Below are various resources that are available to the namespace. You can choose to limit consumption of any or all of these. This is an optional step.

Resource	Limit	Unit
CPU	No limit	MHz
Memory	No limit	MB
Storage	50	GB
vk8s-block-silver	20	GB
vk8s-vvol-silver	10	GB
vk8s-nfs-silver	No limit	MB

[CANCEL] [OK]

After following the previous steps, the storage limit should be visible in the vSphere UI. Note the global limit of 50 GB from the previous steps is also still applied to the vSphere Namespace.



The screenshot shows the vSphere UI for the 'devops' namespace. The 'Storage' tab is selected, displaying 'Storage Policies' with a table of limits:

Storage Policy	Limit
vk8s-block-silver	20 GB Limit
vk8s-vvol-silver	10 GB Limit

Below this table, the 'Capacity and Usage' section shows the global storage limit for the namespace:

Resource	Limit
CPU	No limit
Memory	No limit
Storage	50 GB Limit

**Figure 13. Viewing individual per storage limits in a vSphere Namespace**

Using the **kubectl describe namespace** command, the same 50 GB global storage limit is visible as well as the 20 GB and 10 GB individual per storage limits for block and vVol respectively.

```
C:\>kubectl describe namespace devops
Name:          devops
Labels:        vSphereClusterID=domain-c8
Annotations:   ls_id-0: cea4a565-dfc7-4ee2-a79c-137f9723a86c
               ncp/extpoolid: domain-c8:c2fdf804-b0a0-4bcb-99be-9dab04afa64f-
               ippool-100-88-145-225-100-88-145-254
               ncp/router_id: t1_556a87e1-2308-4992-8418-1445bd1b4dd5_rtr
               ncp/snat_ip: xxx.xxx.xxx.227
               ncp/subnet-0: 10.244.0.32/28
               vmware-system-resource-pool: resgroup-3033
               vmware-system-resource-pool-cpu-limit:
               vmware-system-resource-pool-memory-limit:
               vmware-system-vm-folder: group-v3034
Status:        Active
```

## Resource Quotas

Name:	devops	
Resource	Used	Hard
-----	---	---
requests.storage	0	50Gi

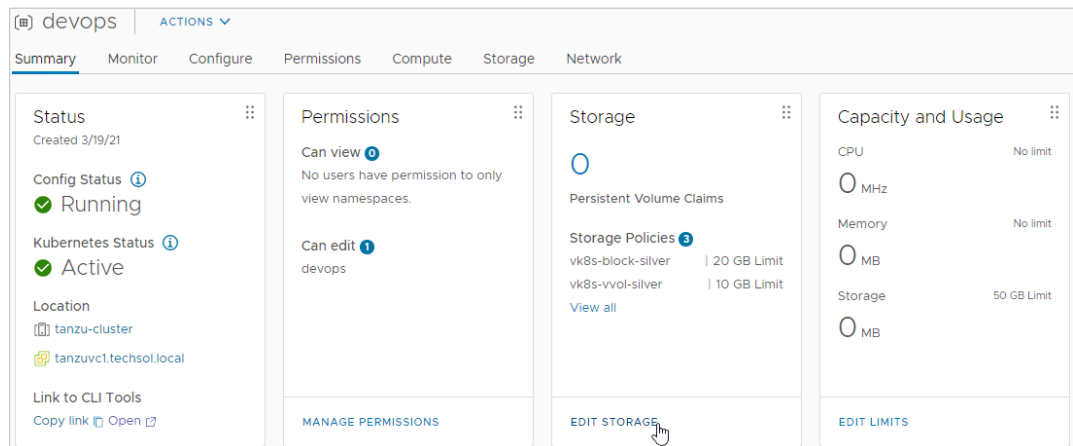
Name:	devops-	
storagequota		
Resource	Used	Hard
-----	---	---
vk8s-block-silver.storageclass.storage.k8s.io/requests.storage	0	20Gi
vk8s-nfs-silver.storageclass.storage.k8s.io/requests.storage	0	
9223372036854775807		
vk8s-vvol-silver.storageclass.storage.k8s.io/requests.storage	0	10Gi

No LimitRange resource.

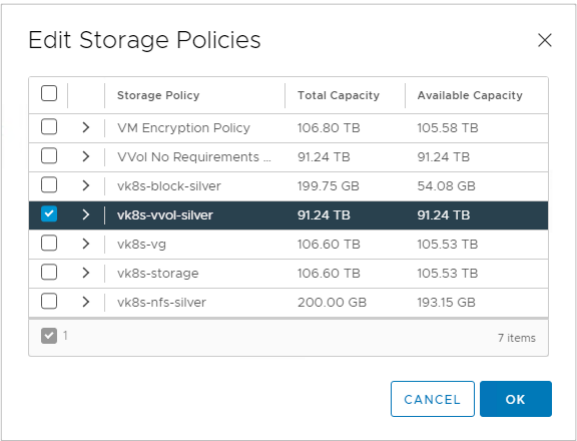
## Removing storage in a vSphere Namespace

When a storage class is no longer in use or needed, it can be removed from the vSphere Namespace. To remove storage from a vSphere Namespace, follow these steps:

1. In the vSphere UI, go to Workload Management. Go into to the vSphere Namespace, and click **Edit Storage**.



2. Clear the checkbox for the storage to be removed. Storage that should remain should be checked. Click **OK**.



After following the previous steps, two Storage Policies have been removed and one storage Policy remains. The storage limits added in previous step remain in place.

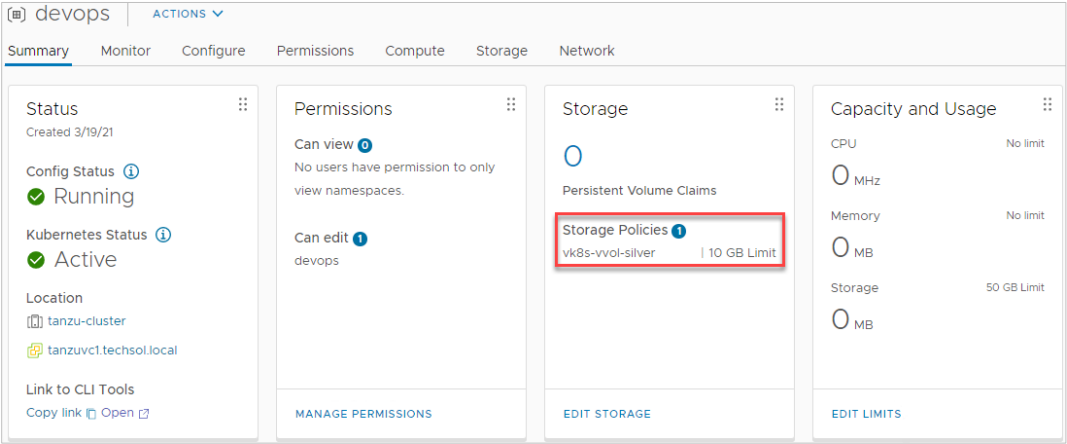


Figure 14. Storage Policies available for use as persistent storage in the vSphere Namespace

## Persistent volumes

A persistent volume is a storage resource that is requested by DevOps team through a persistent volume claim. A persistent volume is used to maintain stateful data for containerized applications. This section will demonstrate the process of provisioning and deleting a dynamic persistent volume backed by PowerStore storage.

### Provision a persistent volume

To provision a dynamic persistent volume, follow these steps:

1. Create the persistent volume claim in a text editor and save as a .yaml file. In this example, 5 GB is being requested from the **vk8s-vvol-silver** storage class.

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: my-pvc
spec:
  storageClassName: vk8s-vvol-silver
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 5Gi
```

2. Log in as a DevOps user and apply the persistent volume claim .yaml using the Kubernetes CLI Tools.

```
C:\>kubectl apply -f C:\Users\jboche\Downloads\my-pvc.yaml
persistentvolumeclaim/my-pvc created
```

Once the persistent volume claim has been applied, it will be visible in the vSphere UI along with storage capacity it consumes.

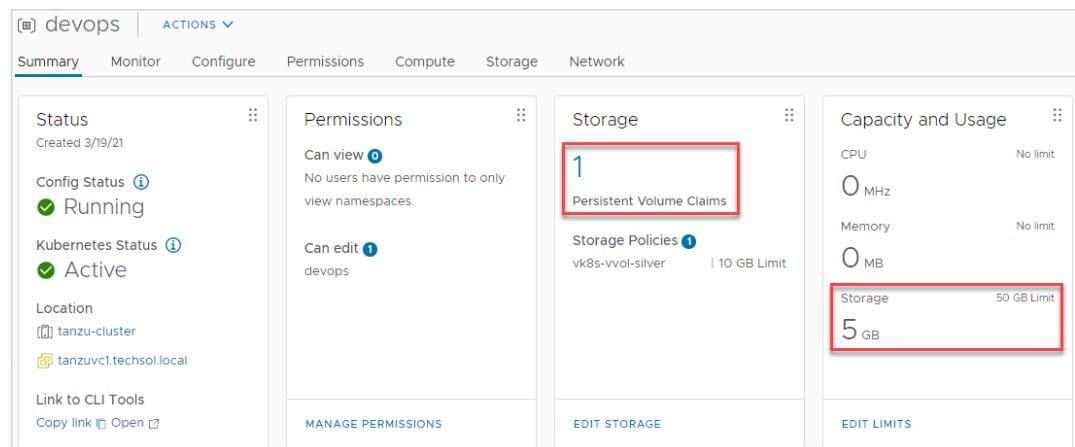


Figure 15. Viewing a persistent volume claim of 5 GB in the vSphere UI

The **kubectl get pvc** command should return the new persistent volume claim of 5Gi using vVol storage and a status of **bound**.

```
C:\>kubectl get pvc
NAME          STATUS    VOLUME                                     CAPACITY   ACCESS
MODES        STORAGECLASS   AGE
my-pvc       Bound       pvc-972f2d9a-8887-4645-8a64-ce9151360c4a   5Gi        RWO
vk8s-vvol-silver 67s
```

---

**Note:** If the new persistent volume claim is not shown or is not bound, check the vCenter Server for errors.

---

The **kubectl describe namespace** command will reflect the persistent volume claim of 5Gi along with the storage limits imposed in the previous section.

```
C:\>kubectl describe namespace devops
Name:          devops
Labels:         vSphereClusterID=domain-c8
Annotations:    ls_id-0: cea4a565-dfc7-4ee2-a79c-137f9723a86c
                ncp/extpoolid: domain-c8:c2fdf804-b0a0-4bcb-99be-
                9dab04afa64f-ippool-100-88-145-225-100-88-145-254
                ncp/router_id: t1_556a87e1-2308-4992-8418-
                1445bd1b4dd5_rtr
                ncp/snat_ip: xxx.xxx.xxx.227
                ncp/subnet-0: 10.244.0.32/28
                vmware-system-resource-pool: resgroup-3033
                vmware-system-resource-pool-cpu-limit:
                vmware-system-resource-pool-memory-limit:
                vmware-system-vm-folder: group-v3034
Status:        Active
```

#### Resource Quotas

```
Name:          devops
Resource       Used   Hard
-----
requests.storage 5Gi   50Gi
```

```
Name:          devops-
storagequota
Resource       Used   Hard
-----
vk8s-vvol-silver.storageclass.storage.k8s.io/requests.storage 5Gi   10Gi
```

No LimitRange resource.

### Delete a persistent volume claim

Once a persistent volume is no longer in use by the vSphere Pods in the vSphere Namespace, its corresponding persistent volume claim can be removed.

To delete a dynamic persistent volume claim, follow these steps:

1. Using the Kubernetes CLI Tools, log in as a DevOps user and delete the persistent volume claim using the same .yaml file that was used to originally request the persistent volume.

```
C:\>kubectl delete -f C:\Users\jboche\Downloads\my-pvc.yaml
persistentvolumeclaim "my-pvc" deleted
```

2. Use the vSphere UI or the **kubectl describe namespace** command to confirm that the persistent volume claim has been removed.

Once the persistent volume claim has been deleted, it should disappear from the vSphere UI and replenish the storage capacity it was once using.

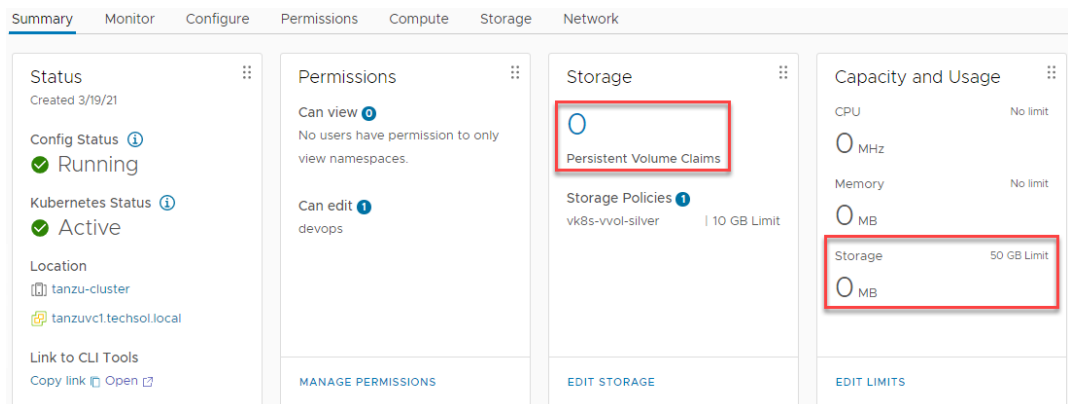


Figure 16. Observing the persistent volume removal in the vSphere UI

Although the persistent volume claim has been deleted and confirmed in the vSphere UI, the first class disk containing stateful data on physical storage remains. The first class disk will be visible in the vSphere UI when examining datastore files and PowerStore Manager if the backing storage was vVols. Assuming the stateful data contained in the first class disk is no longer needed, the last step in the cleanup process is to manually delete the first class disk using the datastore browser in the vSphere UI. If vVols were used for persistent storage, the first class disks can also be removed using PowerStore Manager.

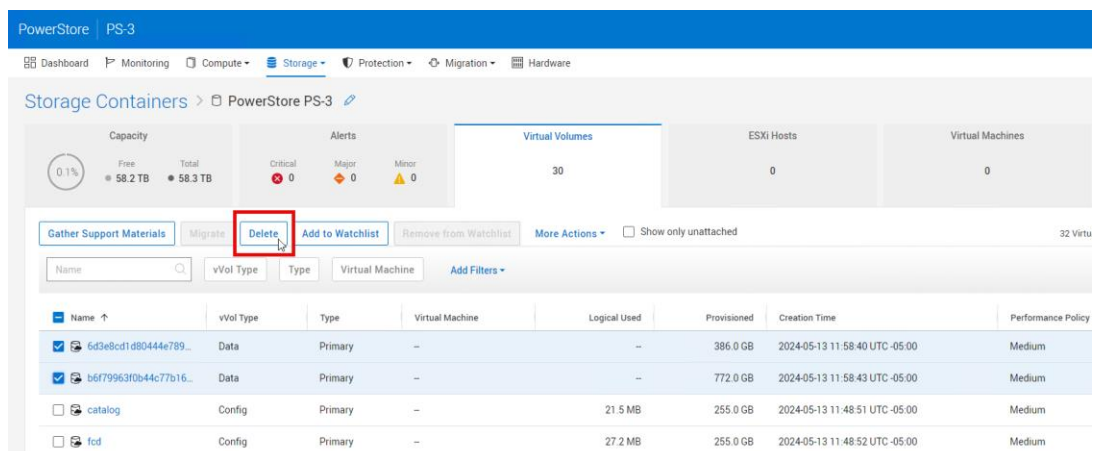


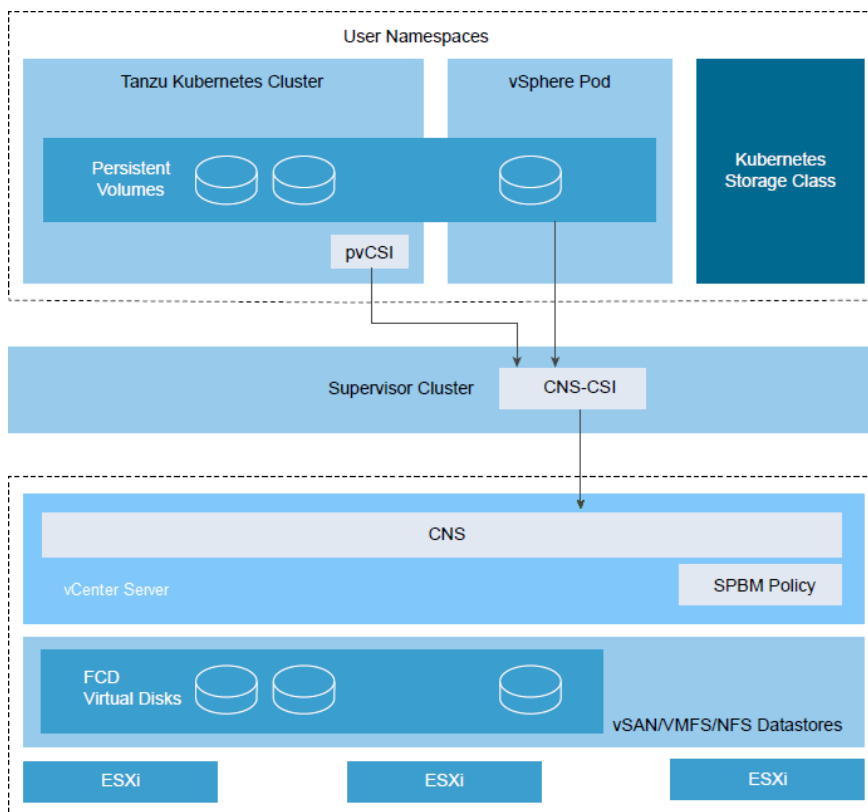
Figure 17. Using PowerStore Manager to remove first class disks once used by persistent storage

## Deploy a Tanzu Kubernetes Cluster

The last section of this document will highlight the deployment of a Tanzu Kubernetes Cluster. A Tanzu Kubernetes Cluster is a full distribution of the open-source Kubernetes container orchestration platform and is deployed on a vSphere with Tanzu Supervisor Cluster by using the Tanzu Kubernetes Grid Service.

Workloads and services can be deployed to Tanzu Kubernetes Clusters using the same tools and methods as with standard Kubernetes clusters. DevOps teams may prefer TKG clusters because they have control over the Kubernetes cluster, including root level access to the control plane and worker nodes. Among other things, this allows DevOps teams to create their own namespaces and maintain the TKG cluster with current Kubernetes versions without having to manage or upgrade the vSphere Supervisor Cluster.

Regarding storage, vSphere with Tanzu uses the vSphere CSI driver to consume storage. TKG clusters consume storage using a separate Paravirtual CSI (pvCSI) driver. The pvCSI is the version of the vSphere CNS-CSI driver modified for Tanzu Kubernetes clusters. The pvCSI resides in the Tanzu Kubernetes cluster and is responsible for all storage-related requests originating from the Tanzu Kubernetes cluster. Storage requests are delivered to the CNS-CSI and then propagate to the CNS in vCenter Server. The pvCSI does not have direct communication with the CNS component. It relies on the CNS-CSI for storage provisioning.



**Figure 18. vSphere CNS-CSI and pvCSI architecture (credit: VMware)**

To deploy a Tanzu Kubernetes Cluster, follow these steps:

1. Open a text editor and construct the .yaml file. There are many community-driven GitHub repositories with example code and demo applications that can be used



instead of creating from scratch. [This one](#) is a good example. In this deployment, the TKG cluster named **tkc-01** will consist of one control plane node and three worker nodes. In addition, each node will consume vVol storage using the **vk8s-vvol-silver** storage policy created earlier.

```
kind: TanzuKubernetesCluster #required parameter
metadata:
  name: tkc-01 #cluster name, user defined
  namespace: devops #supervisor namespace
spec:
  distribution:
    version: v1.16 #resolved kubernetes version
  topology:
    controlPlane:
      count: 1 #number of control plane nodes
      class: guaranteed-small #vmclass for control plane nodes
      storageClass: vk8s-vvol-silver #storageclass for control
plane nodes
    workers:
      count: 3 #number of worker nodes
      class: guaranteed-small #vmclass for worker nodes
      storageClass: vk8s-vvol-silver #storageclass for worker nodes
```

## 2. Use Kubernetes CLI Tools to log in and switch to the **devops** vSphere Namespace.

```
C:\>kubect1 vsphere login --server=https://xxx.xxx.xxx.193 --
vsphere-username devops@vsphere.local --insecure-skip-tls-
verify
```

```
Password:
Logged in successfully.
```

```
You have access to the following contexts:
  xxx.xxx.xxx.193
  devops
```

If the context you wish to use is not in this list, you may need to try logging in again later, or contact your cluster administrator.

To change context, use ``kubect1 config use-context <workload name>``

```
C:\>kubect1 config use-context devops
Switched to context "devops".
```

## 3. Apply the .yaml file to deploy the TKG cluster.

```
C:\>kubect1 apply -f C:\Users\jboche\Downloads\Kubernetes-
master\Kubernetes-master\demo-applications\create-tkc-cluster.yaml
tanzukubernetescluster.run.tanzu.vmware.com/tkc-01 created
```

The deployment process will take several minutes. The progress can be monitored in the vSphere UI shown below. Watch as the control plane node is created, followed by the worker nodes.

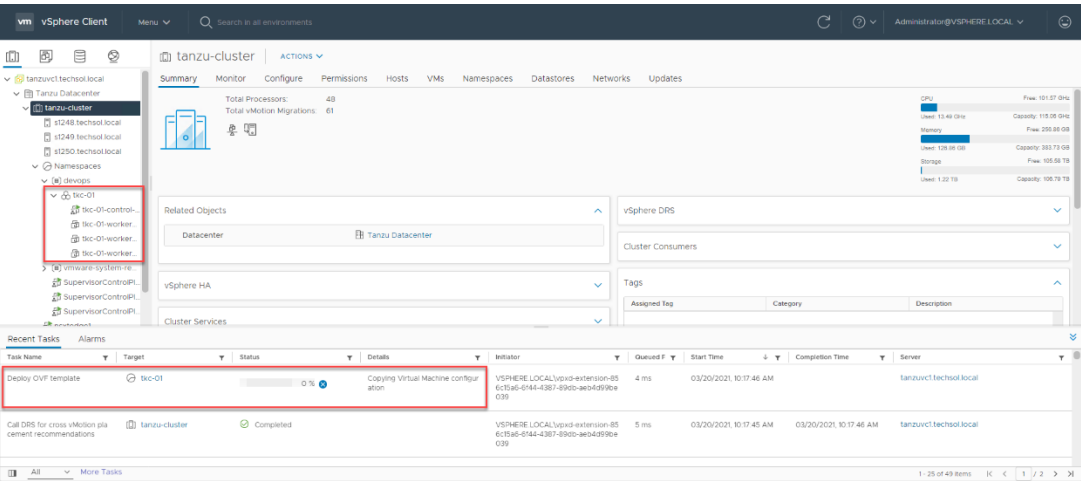


Figure 19. Monitoring the deployment of a TKG cluster in the vSphere UI

The deployment process can also be monitored using the Kubernetes CLI Tools with the **kubectl get events -w** command as shown below. It is not uncommon to see many errors scroll by during the deployment. Errors are normal, and can typically be ignored.

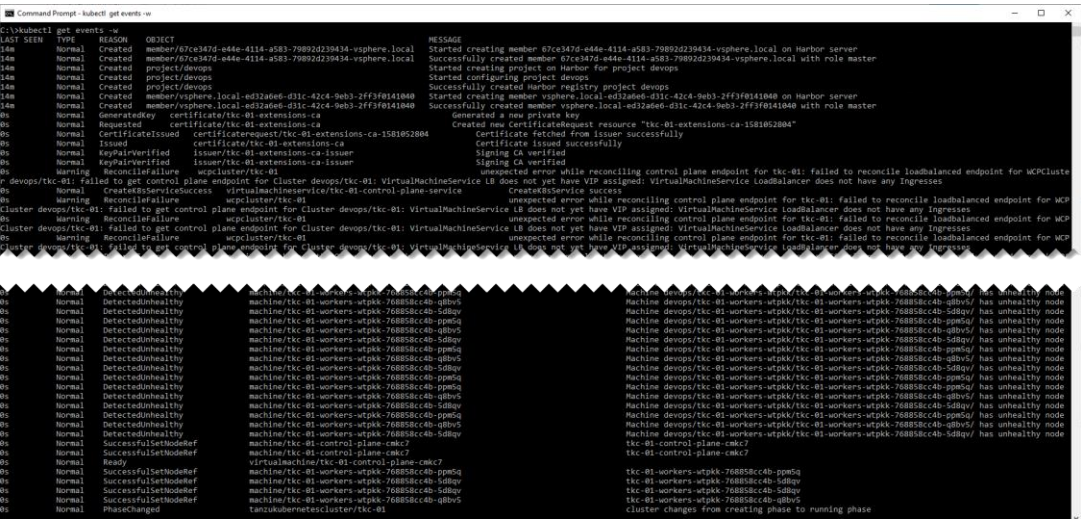
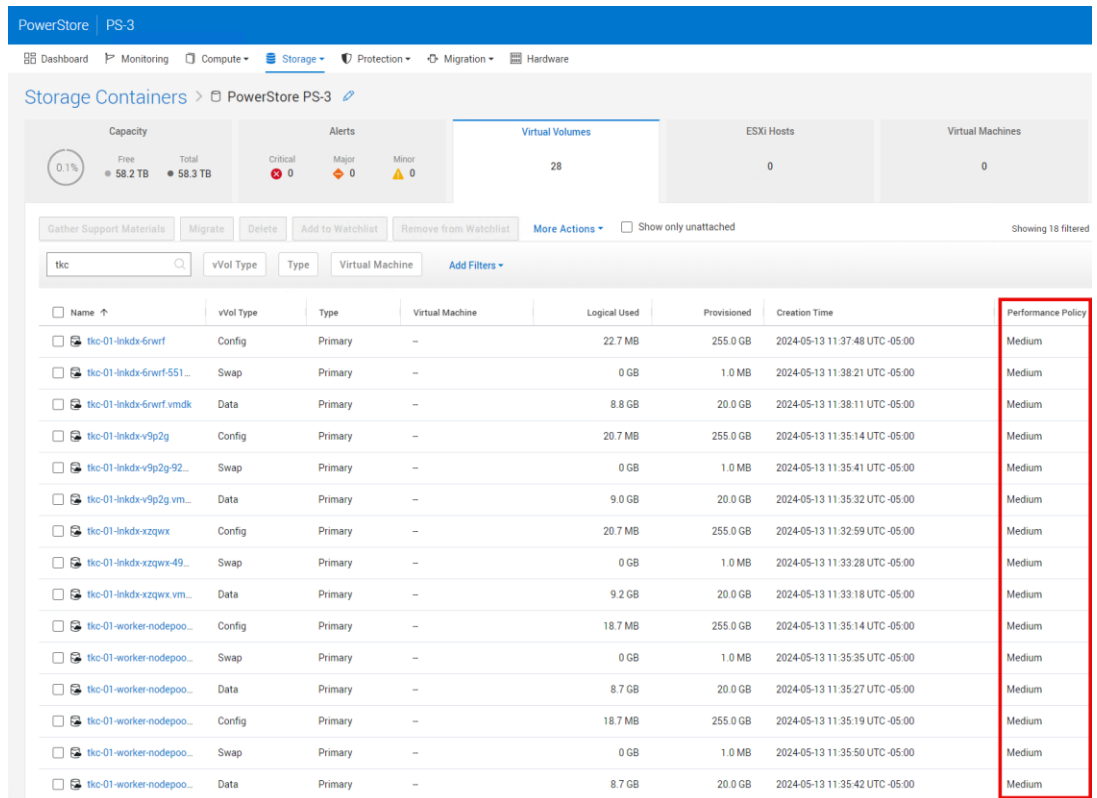


Figure 20. Monitoring the deployment of a TKG cluster using the Kubernetes CLI Tools

The PowerStore Manager UI reflects the deployment of the TKG control plane and worker nodes on vVol storage with an IO Priority of Medium.



**Figure 21. Examining TKG cluster consumption of vVol storage**

After the TKG cluster is deployed, the **devops** user can log in and look around using the Kubernetes CLI Tools.

```
C:\>kubectl vsphere login --server=https://xxx.xxx.xxx.193 --
vsphere-username devops@vsphere.local --insecure-skip-tls-verify --
tanzu-kubernetes-cluster-name tkc-01 --tanzu-kubernetes-cluster-
namespace devops
```

Password:  
Logged in successfully.

You have access to the following contexts:

```
xxx.xxx.xxx.193
devops
tkc-01
```

If the context you wish to use is not in this list, you may need to try logging in again later, or contact your cluster administrator.

To change context, use ``kubectl config use-context <workload name>``

```
C:\>kubectl config use-context tkc-01
Switched to context "tkc-01".
```

```
C:\>kubectl get nodes
```

NAME	STATUS	ROLES	AGE
tkc-01-control-plane-cmkc7 v1.16.14+vmware.1	Ready	master	76m
tkc-01-workers-wtpkk-768858cc4b-5d8qv v1.16.14+vmware.1	Ready	<none>	72m
tkc-01-workers-wtpkk-768858cc4b-ppm5q v1.16.14+vmware.1	Ready	<none>	72m
tkc-01-workers-wtpkk-768858cc4b-q8bv5 v1.16.14+vmware.1	Ready	<none>	72m

```
C:\>kubectl config get-contexts
```

CURRENT	NAME	CLUSTER	AUTHINFO
	xxx.xxx.xxx.193	xxx.xxx.xxx.193	
wcp:xxx.xxx.xxx.193:devops@vsphere.local	devops	xxx.xxx.xxx.193	
wcp:xxx.xxx.xxx.193:devops@vsphere.local	devops		
*	tkc-01	xxx.xxx.xxx.195	
wcp:xxx.xxx.xxx.195:devops@vsphere.local			

```
C:\>kubectl cluster-info
```

Kubernetes master is running at https://xxx.xxx.xxx.195:6443  
 KubeDNS is running at  
 https://xxx.xxx.xxx.195:6443/api/v1/namespaces/kube-  
 system/services/kube-dns:dns/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.

The TKG cluster has been deployed successfully and is ready for management and deployment of containerized applications and services by the DevOps team.

## vSphere with Tanzu and Metro Volume witness

### Introduction

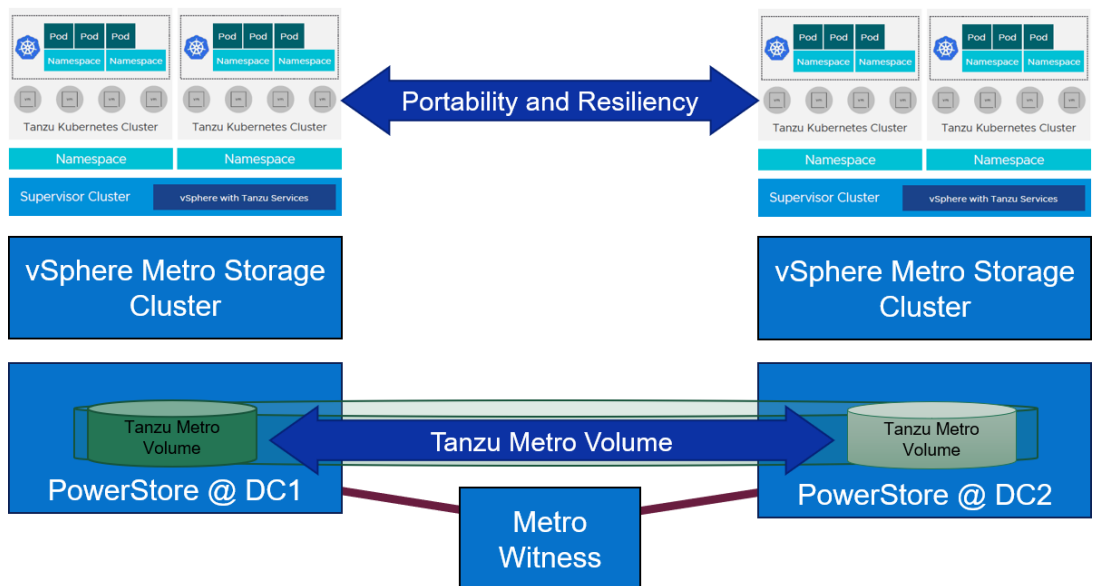
Dell PowerStore Metro Volume provides synchronous replication of spanned block storage volumes across two PowerStore clusters in metro distance for VMFS datastores. The Metro Volume feature provides a path for VMware vSphere Metro Storage Cluster (vMSC) architectures.

The release of PowerStoreOS 3.6 introduces a witness component to the Metro Volume architecture. The functional design of the witness adds more resiliency to Metro Volume deployments and further mitigates risk of split-brain situations. This is accomplished by more intelligent decision making across a wider variety of infrastructure outage scenarios, including unplanned outages.

## A highly available Tanzu infrastructure

With traditional block and vVol storage, an unexpected infrastructure outage can lead to downtime and challenging recovery options for a Tanzu environment. This is because some of the Tanzu components and their corresponding storage components cannot be protected and easily recovered using traditional methods such as snapshots and asynchronous or synchronous replication. Examples of these objects can be found throughout the Workload Management hierarchy: vSphere Tanzu supervisor cluster control plane nodes, pods, Tanzu Kubernetes clusters, image registry, content library, etc.

A vSphere Metro Storage Cluster and Metro Volume with witness architecture significantly enhances both availability and recoverability of Tanzu infrastructure. In the event of a planned or unplanned datacenter or storage outage, containerized workloads and the underlying Tanzu infrastructure components can seamlessly continue running at the remote site where compute, networking, and storage remain healthy and available for consumption. Likewise, NSX Manager and NSX Edge networking components on Metro Volume with witness can also take advantage of this deployment model by remaining highly available during an unplanned outage.



**Figure 22. Metro Volume with witness adds portability and resiliency to Tanzu deployments**

In one working example of this architecture, we ran HammerDB against a containerized SQL Server instance running inside a Tanzu Kubernetes Cluster. After causing an outage of a PowerStore appliance, the HammerDB workload showed continuous availability of the back-end database. With one PowerStore appliance removed (where the back-end database was originally running), the Tanzu Kubernetes Cluster and its workloads shifted seamlessly to the surviving datacenter and PowerStore appliance.

**Note:** The degree of seamlessness and high availability will be influenced by front-end storage presentation. Uniform will yield the best results while non-uniform will incorporate vSphere HA during the recovery along with measurable downtime.

Leveraging Metro Volume storage for Tanzu follows the same processes outlined earlier in the document for block storage. One or more Metro Volumes with witness are created and mapped in PowerStore Manager. In the vSphere Client, the Metro volumes are formatted

as a VMFS datastore, tagged, and assigned a VM Storage Policy. From that point forward, the VM Storage Policy can be selected during or after Tanzu Workload Management enablement for control plane, ephemeral, and image cache storage. The same policy can be used for the Harbor image registry.

**Figure 23. Metro Volume(s) with witness provide highly available stretched storage for control plane, ephemeral, and image cache storage**

A Tanzu Kubernetes Cluster can be deployed on Metro Volume with witness storage by using a storage class which matches the VM Storage Policy that corresponds to Metro Volume storage. Example .yaml is shown below which will deploy a Tanzu Kubernetes Cluster with three control plane nodes and three worker nodes.

```
apiVersion: run.tanzu.vmware.com/v1alpha2
kind: TanzuKubernetesCluster #required parameter
metadata:
  name: tkc-01 #cluster name, user defined
  namespace: devops #supervisor namespace
spec:
  topology:
    controlPlane:
      replicas: 3 #number of control plane nodes
      vmClass: guaranteed-medium #vmclass for control plane nodes
      storageClass: vk8s-mv-storage #storageclass for Metro Volume
      tkr:
        reference:
          name: v1.23.8---vmware.3-tkg.1 #resolved kubernetes version
    nodePools:
      - name: worker-nodepool-a1
        replicas: 3 #number of worker nodes
        vmClass: guaranteed-large #vmclass for worker nodes
        storageClass: vk8s-mv-storage #storageclass for Metro Volume
        tkr:
          reference:
            name: v1.23.8---vmware.3-tkg.1 #resolved kubernetes version
```

## VMware Tanzu Kubernetes Grid

Up to this point, this paper has covered vSphere with Tanzu (also known as TKGs) deployed on PowerStore . vSphere with Tanzu is a fine option for organizations who want to deploy and manage containerized workloads in a Kubernetes runtime that is deeply integrated with vSphere.

VMware also offers and supports Tanzu Kubernetes Grid (also known as TKGm). Tanzu Kubernetes Grid provides a consistent, upstream-compatible implementation of Kubernetes that is tested, signed, and supported by VMware. Tanzu Kubernetes Grid instances include the management cluster, the deployed Tanzu Kubernetes clusters, and the shared and in-cluster services that you configure. Tanzu Kubernetes Grid may be deployed to IaaS providers such as Azure and Amazon EC2 and for different failure domains such as AWS us-east-2, AWS us-west-2, and so on. TKG is a multicloud offering.

Like vSphere with Tanzu, TKG can also be deployed to vSphere infrastructure while vSphere with Tanzu is not enabled. This means that TKG can be deployed using PowerStore for back-end storage.

## Conclusion

Dell PowerStore provides various storage types that are automatically tuned for performance. Dell PowerStore is a great storage and compute platform for vSphere with Tanzu and Tanzu Kubernetes Grid clusters using the vSphere CSI driver.

## Bibliography

Burns, B., Beda, J., & Hightower, K. (2019). *Kubernetes Up & Running*. Sebastopol: O'Reilly.

## Appendix: Additional resources

### Technical support and resources

[Dell.com/support](https://dell.com/support) is focused on meeting customer needs with proven services and support.

[Storage and data protection technical white papers and videos](#) provide expertise that helps to ensure customer success with Dell storage and data protection products.

The [PowerStore Info Hub](#) provides detailed documentation on how to install, configure, and manage PowerStore systems.

PowerStore resources:

- [Dell PowerStore: Best Practices Guide](#)
- [Dell PowerStore: Virtualization Integration](#)
- [Dell PowerStore Virtualization Infrastructure Guide](#)
- [Dell PowerStore: Metro Volume](#)
- [Dell EMC PowerProtect Data Manager protecting VMware Tanzu Kubernetes Clusters](#)
- [PowerStore CSI driver on GitHub](#)
- [Dell Technologies CSI drivers](#)

### VMware support

For VMware support, see the following resources:

- [vSphere with Tanzu Configuration and Management](#)
- [VMware vSphere with Tanzu Release Notes](#)
- [Architecture and Design for a vSphere with Tanzu Workload Domain](#)
- [VMware Validated Design Documentation](#)
- [VMware Tanzu Edge Solution Architecture with ROBO Topology](#)
- [vSphere CSI driver on GitHub](#)
- [Kubernetes vSphere CSI Driver](#)
- [VMware.com](#)
- [VMware support](#)
- [Education and training](#)
- [Online documentation](#)
- [VMware communities](#)