Dell PowerStore: VMware vSphere with Tanzu and TKG Clusters

April 2022

H18731.1

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.

Dell Technologies

Copyright

Contents

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)

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	Description
April 2021	Initial release
April 2022	• Added PowerStore X model support • Updated template

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 in two product models: PowerStore T models and PowerStore X models.

- PowerStore T appliances are bare-metal, unified storage arrays that can service block, file, and VMware vSphere Virtual Volumes (vVols) resources along with numerous data services and efficiencies.
- PowerStore X appliances enable running applications directly on the appliance through the AppsON capability. A native VMware ESXi layer runs embedded applications alongside the PowerStore operating system, all in the form of virtual machines. This feature adds to the traditional storage functionality of PowerStore X model appliances and supports serving external block and vVol storage to servers with FC and iSCSI.

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.

Term	Definition
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.
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 T or PowerStore X appliances can be clustered by adding appliances as required.

Term	Definition
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.
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 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.
PowerStore X model	Container-based storage system that runs inside a virtual machine that is deployed on a VMware hypervisor. Besides offering block-optimized workloads, PowerStore also allows users to deploy applications directly on the array.
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 OS from VMware and are built, signed, and supported by VMware.

Term	Definition
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.
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, the PowerStore T model can be deployed in block optimized mode.
- If vSphere with Tanzu will be using NFS storage, the PowerStore T model should be deployed in unified mode.

The PowerStore X model will natively use vVol storage. 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 EMC 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.

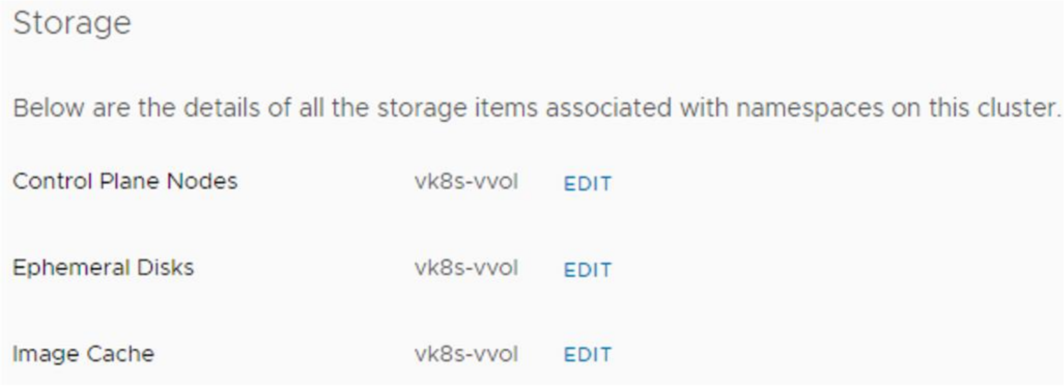
Note: As of PowerStore version 2.1.1, a single PowerStore X appliance deploys two vSphere 7.0 Update 3 ESXi hosts in a cluster. Combining two or more PowerStore X appliances in a PowerStore Cluster meets the minimum vSphere with Tanzu host infrastructure and storage requirements.

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 polices and tags with PowerStore will be covered in an upcoming section.



Storage

Below are the details of all the storage items associated with namespaces on this cluster.

Control Plane Nodes	vk8s-vvol	EDIT
Ephemeral Disks	vk8s-vvol	EDIT
Image Cache	vk8s-vvol	EDIT

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

Virtual machine	Nodes	Total vCPUs	Total memory	Total storage
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
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.

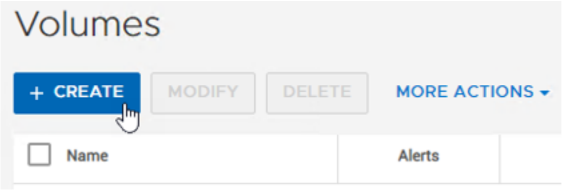
Block volumes

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.

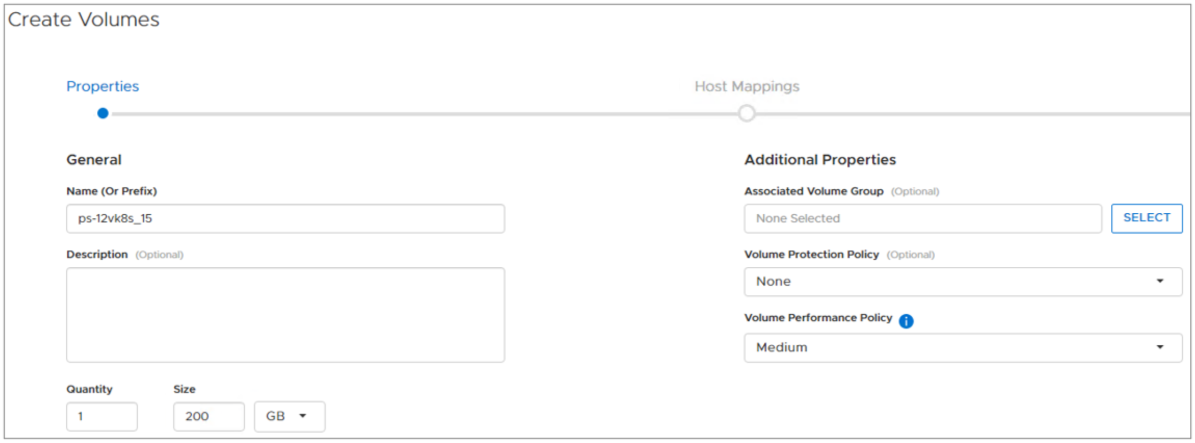
Note: If PowerStore will only be serving block volumes, consider installing PowerStore in block-optimized mode, which disables the NAS and vVol capabilities. This mode can increase the scale or performance of PowerStore since it can devote additional CPU and memory that is no longer needed for file.

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

General

Name (Or Prefix): ps-12vk8s_15

Description (Optional):

Quantity: 1

Size: 200 GB

Additional Properties

Associated Volume Group (Optional): None Selected

Volume Protection Policy (Optional): None

Volume Performance Policy: Medium

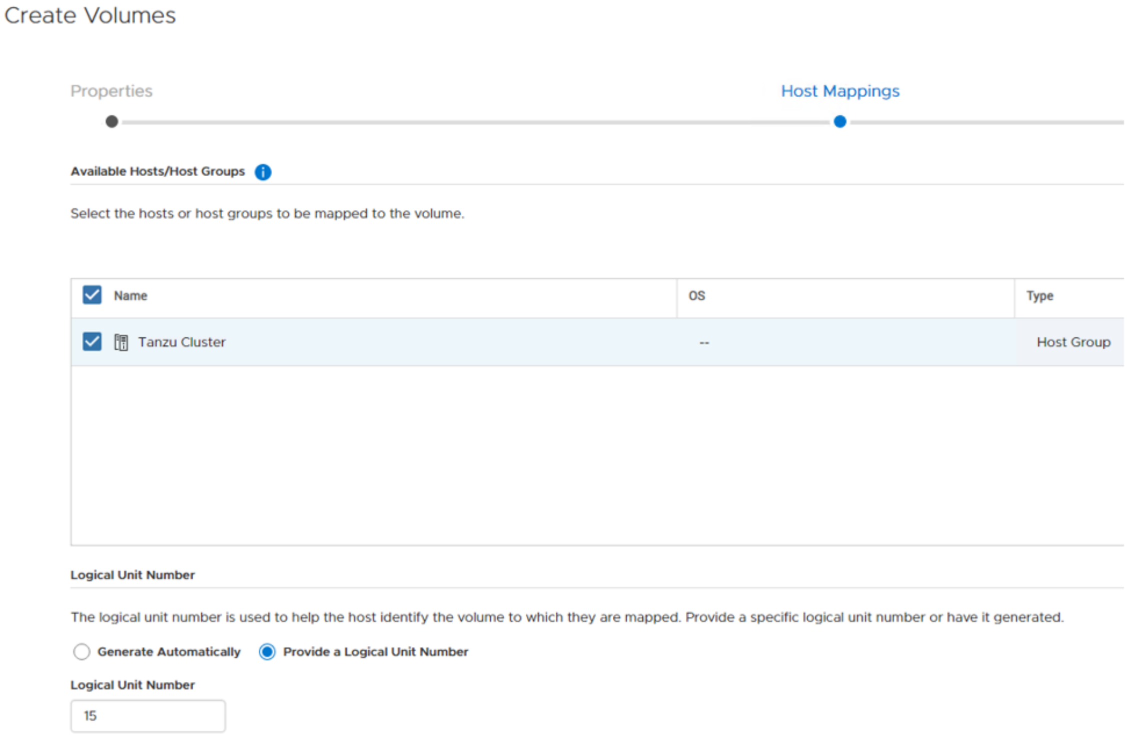
The volume performance policy determines the relative performance priority of the volume using a share-based system. Available options are Low, Medium (default), and High. The volume performance policy can be tied to a tag and a VM Storage Policy and used as a storage tiering mechanism. For example,

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.



Create Volumes

Available Hosts/Host Groups

Select the hosts or host groups to be mapped to the volume.

☑ Name	OS	Type
☑ Tanzu Cluster	--	Host Group

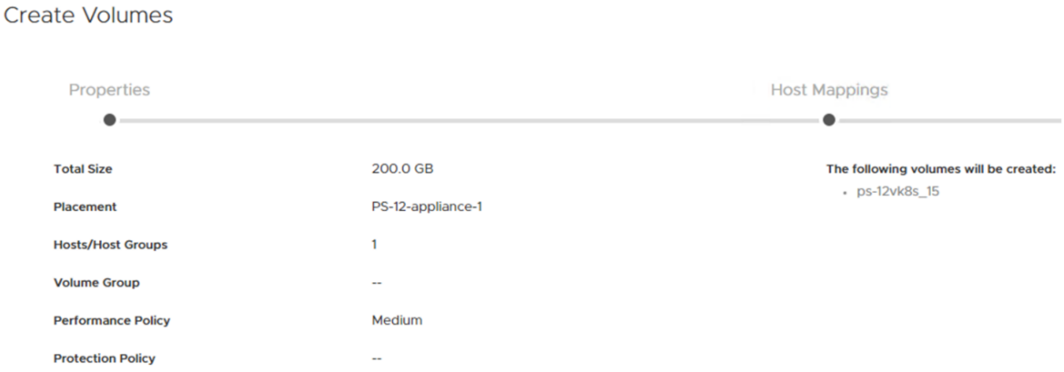
Logical Unit Number

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.



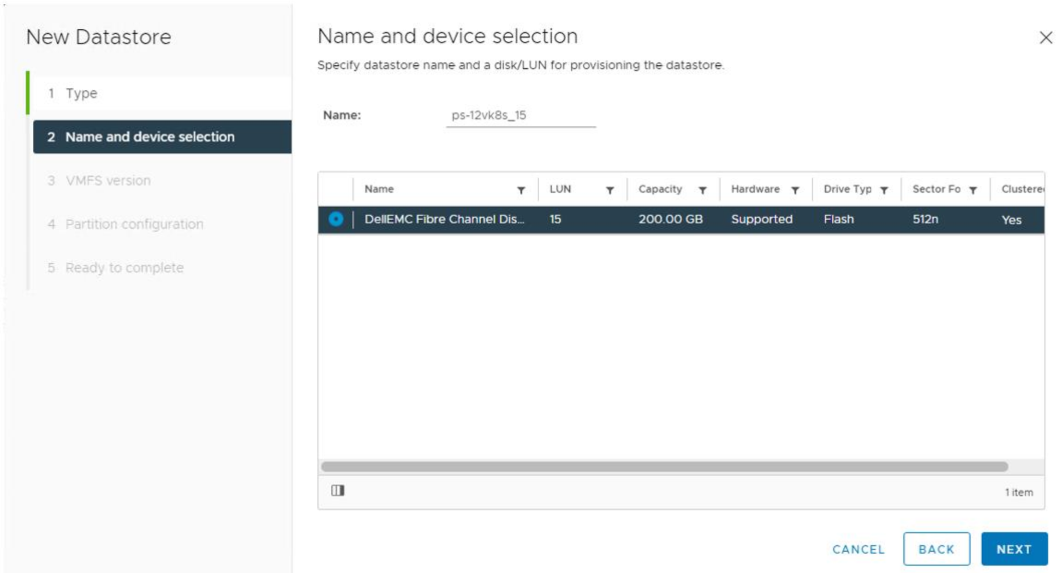
Create Volumes

Total Size	200.0 GB
Placement	PS-12-appliance-1
Hosts/Host Groups	1
Volume Group	--
Performance Policy	Medium
Protection Policy	--

The following volumes will be created:

- ps-12vk8s_15

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



New Datastore

Name and device selection

Specify datastore name and a disk/LUN for provisioning the datastore.

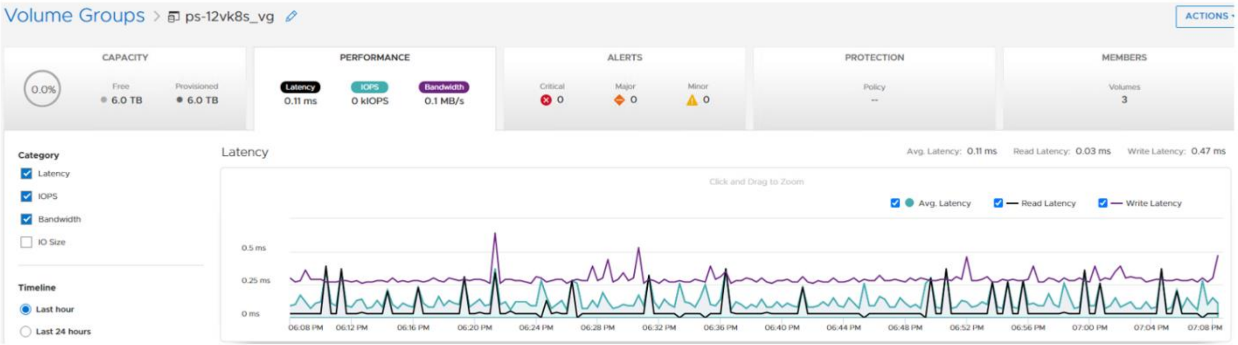
Name: ps-12vk8s_15

Name	LUN	Capacity	Hardware	Drive Typ	Sector Fo	Clustere
DellEMC Fibre Channel Dis...	15	200.00 GB	Supported	Flash	512n	Yes

1 item

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.



Volume Groups > ps-12vk8s_vg

CAPACITY: 0.0% Free 6.0 TB Provisioned 6.0 TB

PERFORMANCE: Latency 0.11 ms IOPS 0 kIOPS Bandwidth 0.1 MB/s

ALERTS: Critical 0 Major 0 Minor 0

PROTECTION: Policy --

MEMBERS: Volumes 3

Category

- ☑ Latency
- ☑ IOPS
- ☑ Bandwidth
- □ IO Size

Timeline

- ● Last hour
- ○ Last 24 hours

Latency

Avg. Latency: 0.11 ms Read Latency: 0.03 ms Write Latency: 0.47 ms

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 replication for data protection. One exception might be Persistent Volumes. Persistent Volumes are backed by independent First Class Disks (FCD) which may contain data that should be protected. For more information about protecting VMware Tanzu Clusters, refer to the document Dell EMC PowerProtect Data Manager protecting VMware Tanzu Kubernetes Clusters.

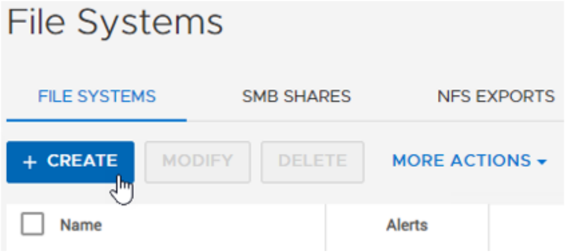
NFS volumes

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.

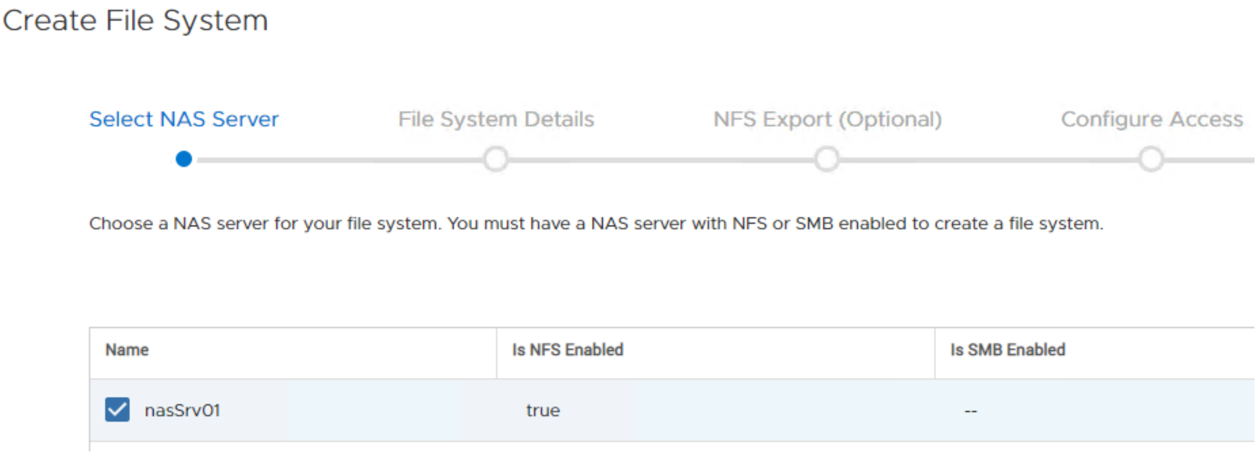
Note: A NAS Server must be created before a File System and NFS Export can be created. See the PowerStore documentation for the process to create a NAS Server.

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

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



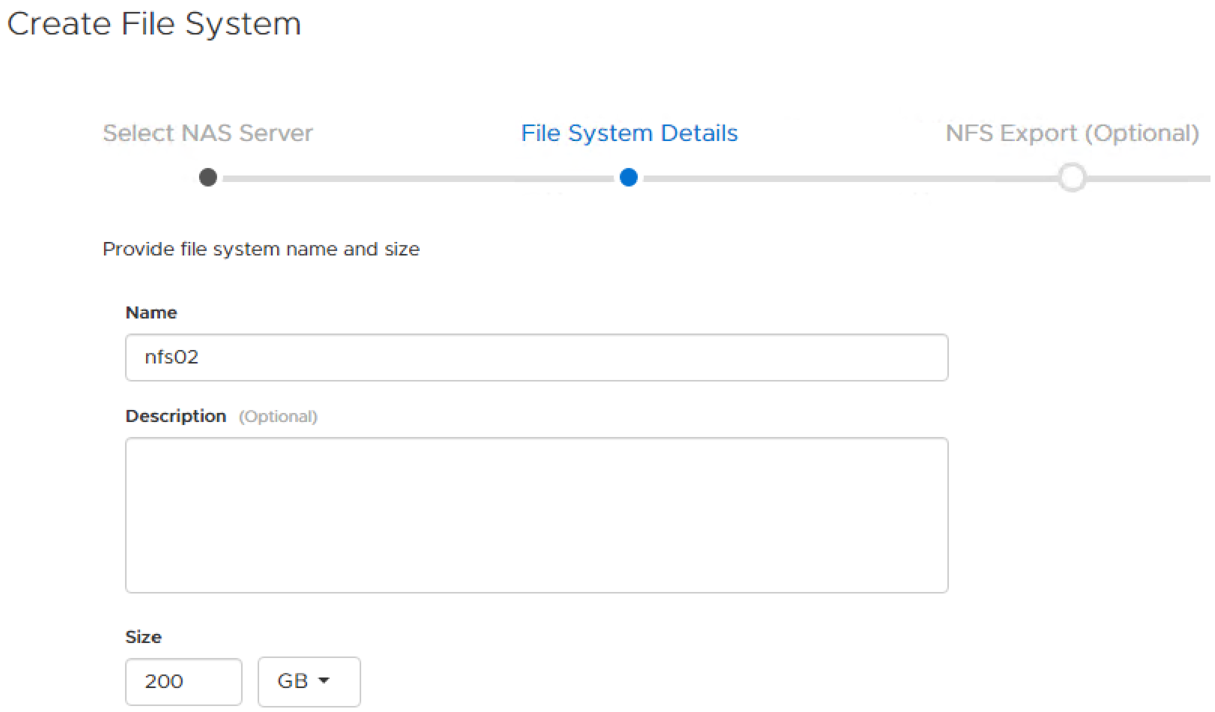
2. Select the NAS Server and click Next.



Choose a NAS server for your file system. You must have a NAS server with NFS or SMB enabled to create a file system.

Name	Is NFS Enabled	Is SMB Enabled
☑ nasSrv01	true	--

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



Create File System

Provide file system name and size

Name

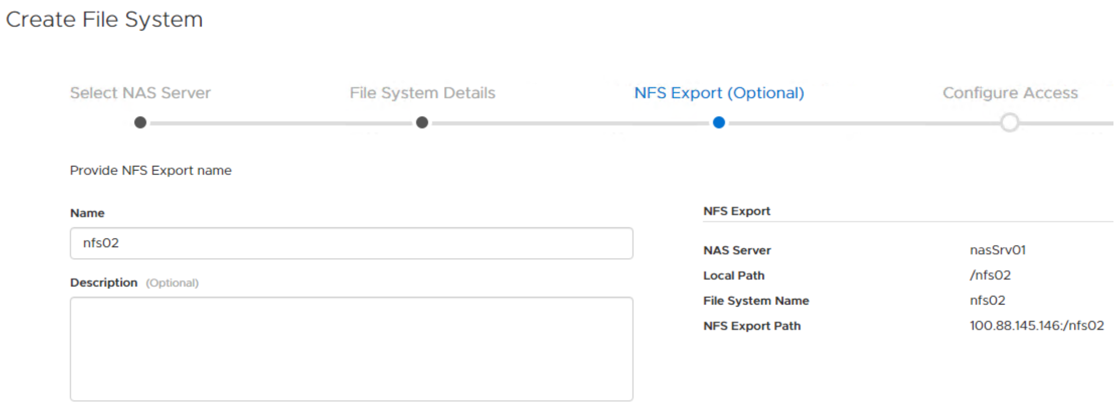
nfs02

Description (Optional)

Size

200 GB

4. Specify an NFS Export name. Note the NFS Export details as they will be needed later. Click Next.



Create File System

Provide NFS Export name

Name

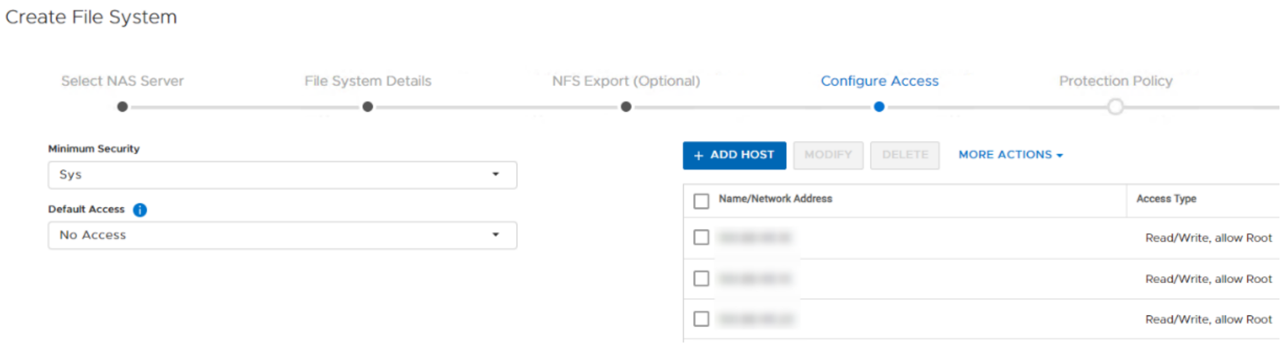
nfs02

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.



Create File System

Minimum Security

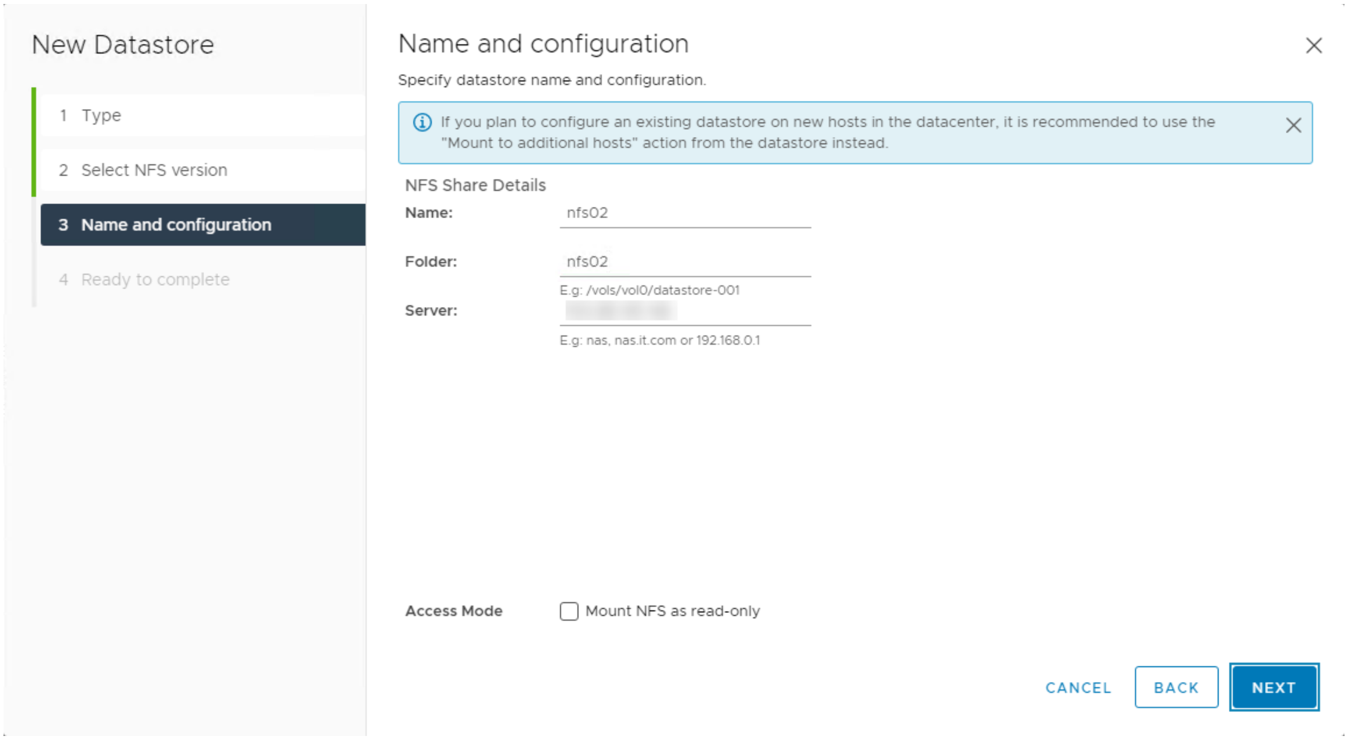
Sys

Default Access

No Access

□ Name/Network Address	Access Type
□ [redacted]	Read/Write, allow Root
□ [redacted]	Read/Write, allow Root
□ [redacted]	Read/Write, allow Root

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.



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

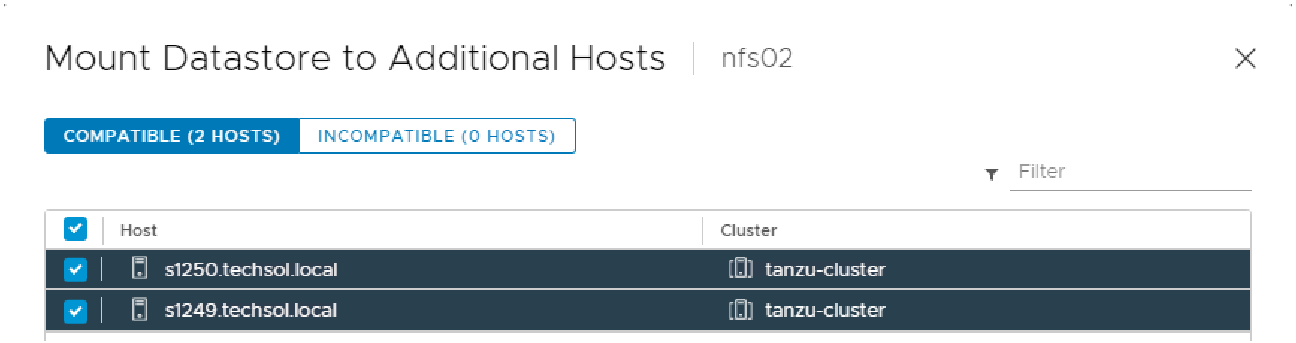
Name: nfs02

Folder: nfs02
E.g: /vols/vol0/datastore-001

Server: [redacted]
E.g: nas, nas.it.com or 192.168.0.1

Access Mode ☐ Mount NFS as read-only

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



Mount Datastore to Additional Hosts | nfs02

☑	Host	Cluster
☑	s1250.techsol.local	tanzu-cluster
☑	s1249.techsol.local	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/csidriver/.

The PowerStore T model and PowerStore X model 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 EMC PowerStore: Virtualization Integration.

Note: The PowerStore X model deployment automatically registers the VASA provider with the vCenter Server and the vVol datastore is automatically created.

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

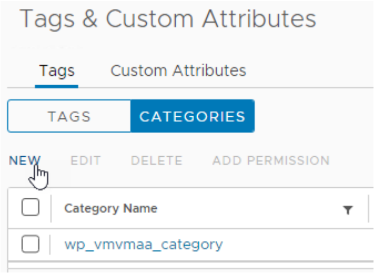
After PowerStore model T or model X has been deployed, VM Storage Policies must be created and tied to the underlying storage. VM Storage Policies leverage VMFS block and file datastores by using tags. VM Storage Policies leverage vVol storage through the VASA provider. The following sections will provide 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. For a PowerStore T model, tags are applied VMFS block and file datastores.

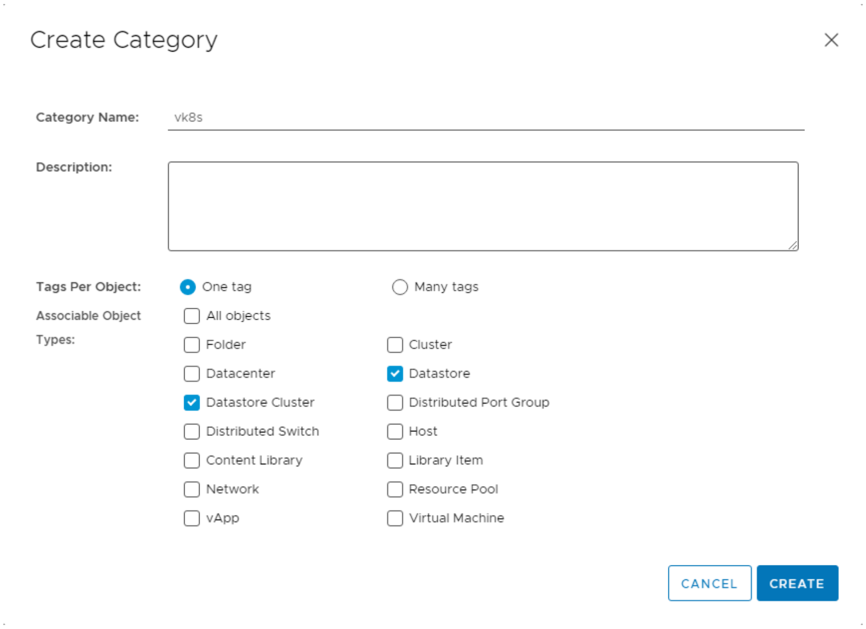
To create a tag category, follow these steps:

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



□	Category Name
□	wp_vmvmaa_category

3. Assign a Category Name. For Associable Object Types choose Datastore and optionally Datastore Cluster if Datastore Clusters will be used. Click Create.



Category Name: vk8s

Description:

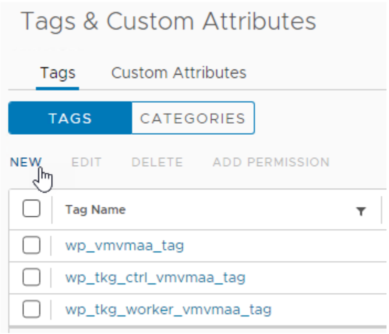
Tags Per Object: ● One tag ○ Many tags

Associable Object Types:
□ All objects
□ Folder □ Cluster
□ Datacenter ☑ Datastore
☑ Datastore Cluster □ Distributed Port Group
□ Distributed Switch □ Host
□ Content Library □ Library Item
□ Network □ Resource Pool
□ vApp □ Virtual Machine

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.

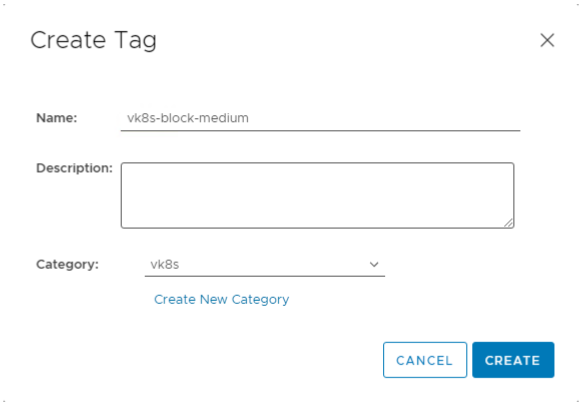
To create a tag, follow these steps:

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



□	Tag Name
□	wp_vmvmaa_tag
□	wp_tkg_ctrl_vmvmaa_tag
□	wp_tkg_worker_vmvmaa_tag

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



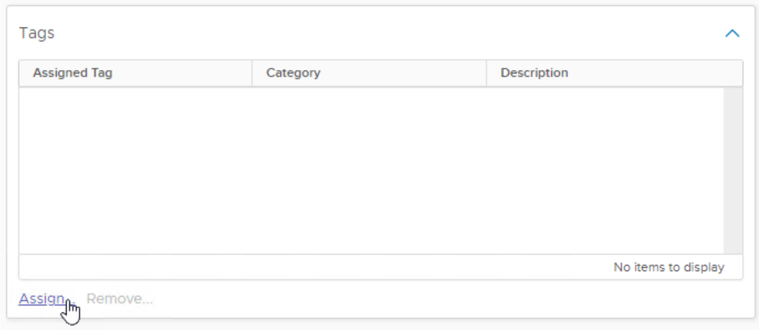
Name: vk8s-block-medium

Description:

Category: vk8s

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

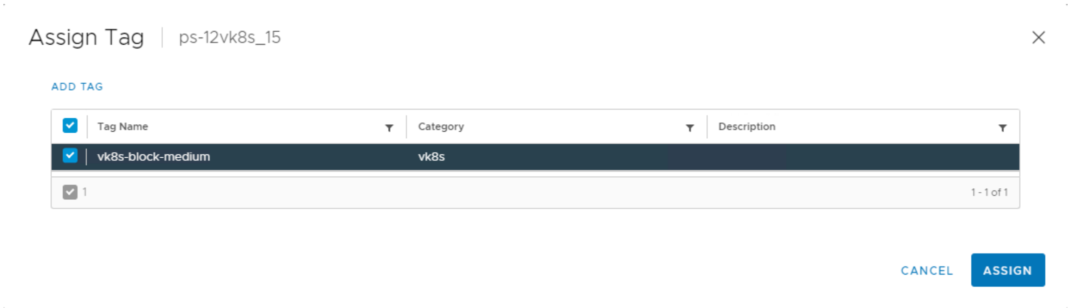
1. In the vSphere UI, go to Storage.
2. Identify the datastore to be tagged and left click on it in the inventory tree.
3. Under Tags, click Assign.



Assigned Tag	Category	Description

No items to display

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



	Tag Name	Category	Description
☑	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 are deployed.

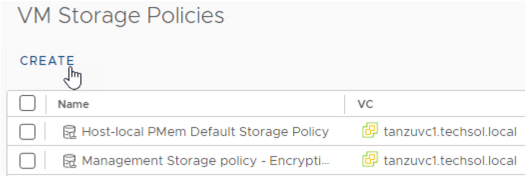
Creating a VM Storage Policy is the final configuration needed before vSphere with Tanzu can be deployed and examples will be provided in the following sections.

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

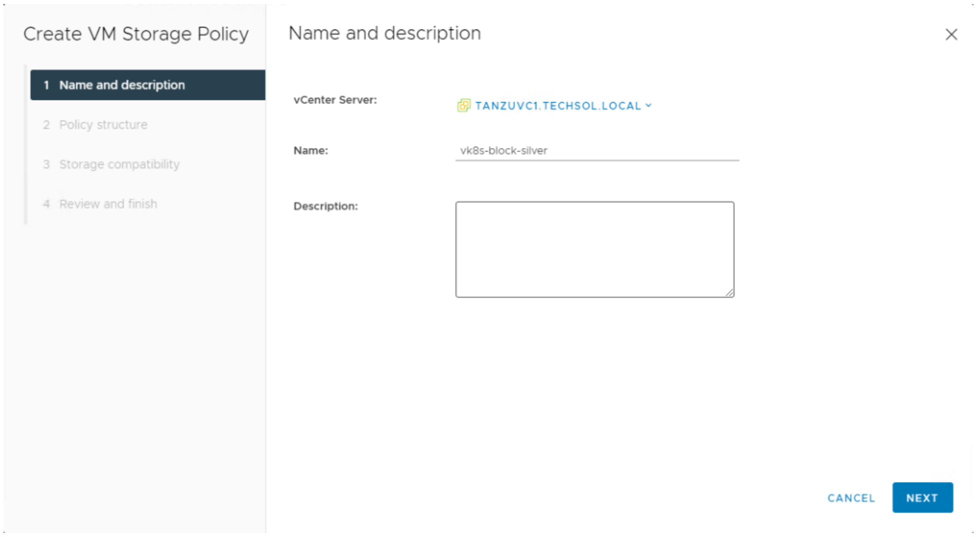
To create a VM Storage Policy for block or NFS storage on the PowerStore T model, follow these steps:

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



□	Name	VC
□	Host-local PMem Default Storage Policy	tanzuvc1.techsol.local
□	Management Storage policy - Encrypti...	tanzuvc1.techsol.local

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

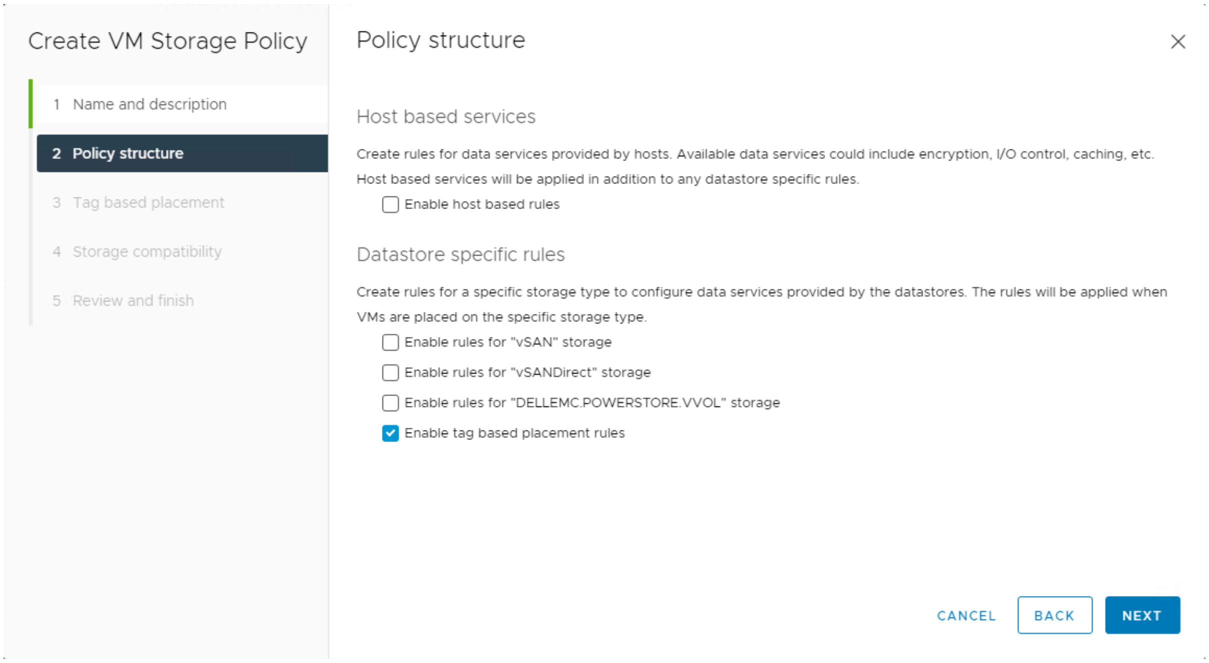


vCenter Server: TANZUVC1.TECHSOL.LOCAL

Name: vk8s-block-silver

Description:

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

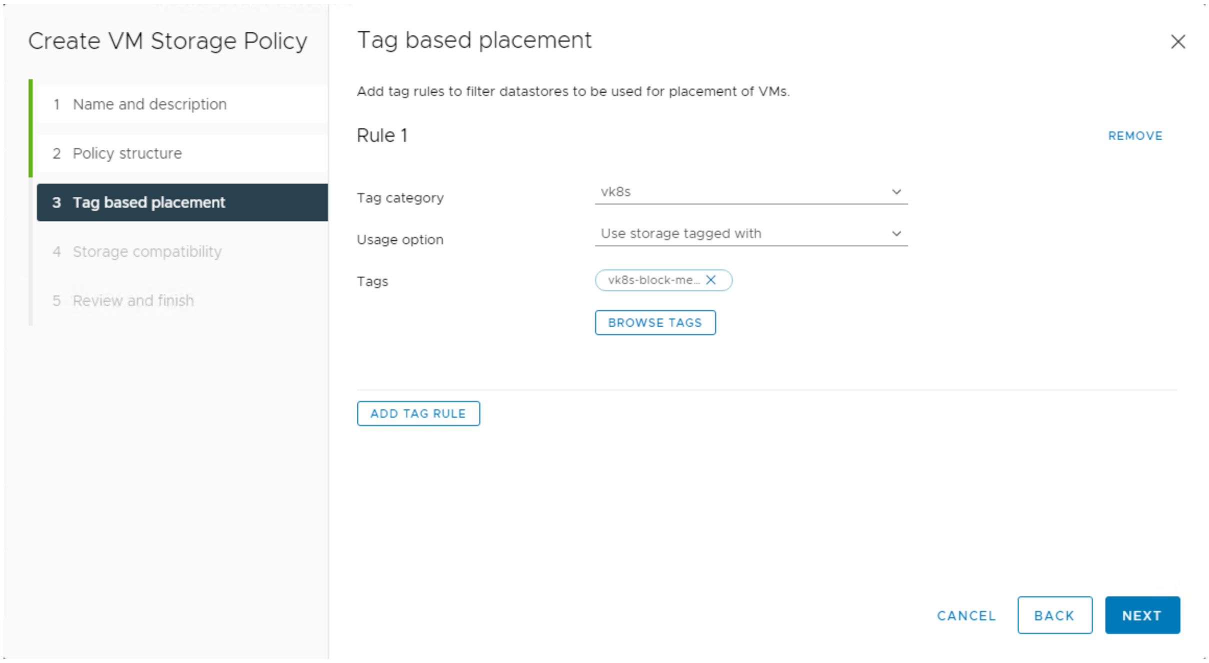
Datastore specific rules

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 "DELLEMC.POWERSTORE.VVOL" storage
☑ Enable tag based placement rules

CANCEL BACK NEXT

5. Choose the Tag category created earlier. For Usage option, choose Use storage tagged with. Click the Browse Tags button and choose an appropriate tag created earlier. In this example, a silver policy and storage class is being created which ties 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

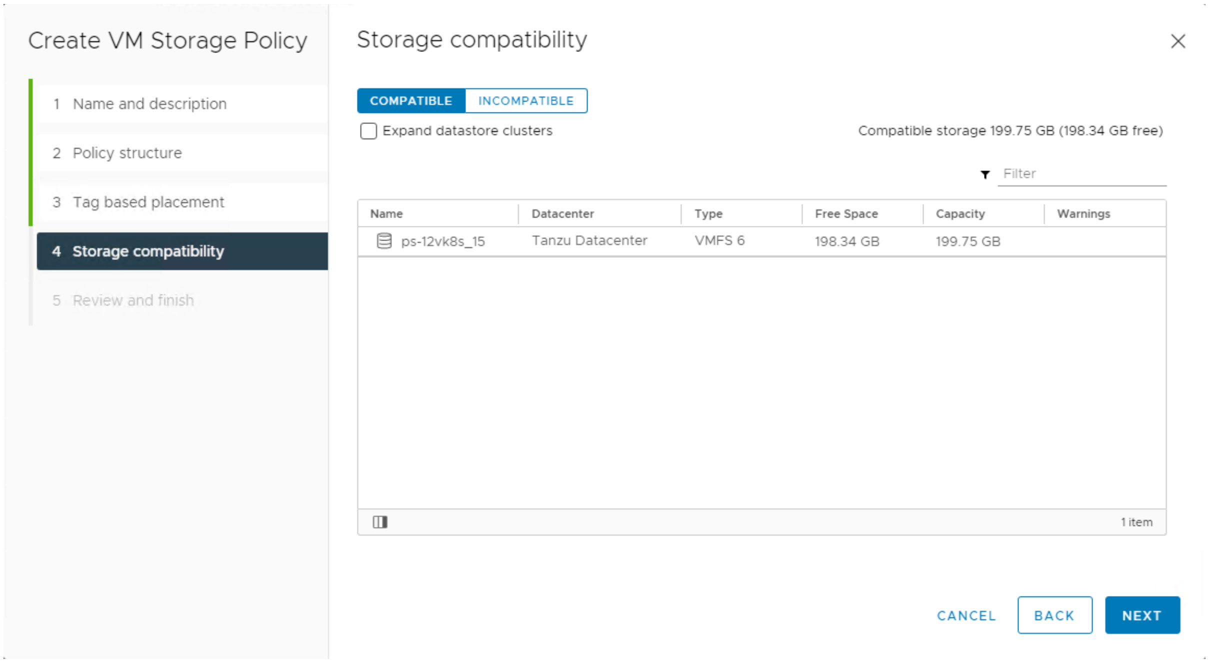
Tag category: vk8s
Usage option: Use storage tagged with
Tags: vk8s-block-me... ×

BROWSE TAGS

ADD TAG RULE

CANCEL BACK NEXT

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



Create VM Storage Policy

Storage compatibility

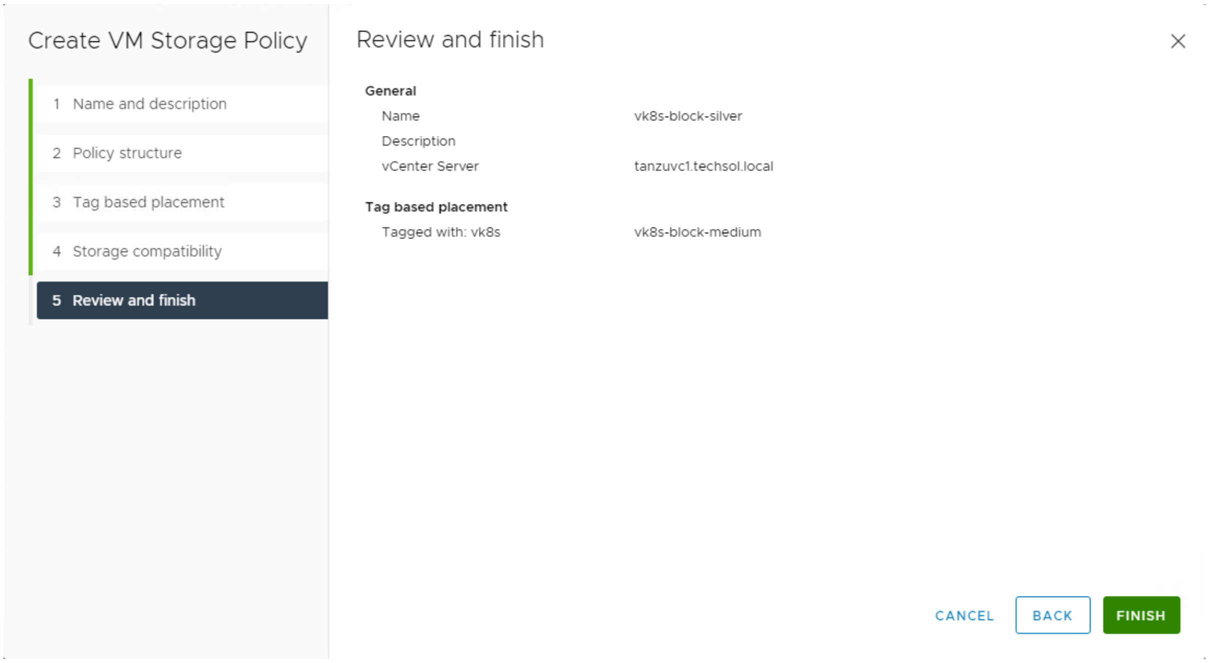
□ Expand datastore clusters

Compatible storage 199.75 GB (198.34 GB free)

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

1 item

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



Create VM Storage Policy

Review and finish

General

Name	vk8s-block-silver
Description	
vCenter Server	tanzuvc1.techsol.local

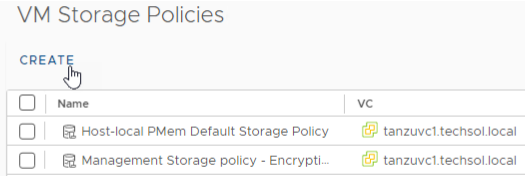
Tag based placement

Tagged with: vk8s	vk8s-block-medium

vVol storage

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

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

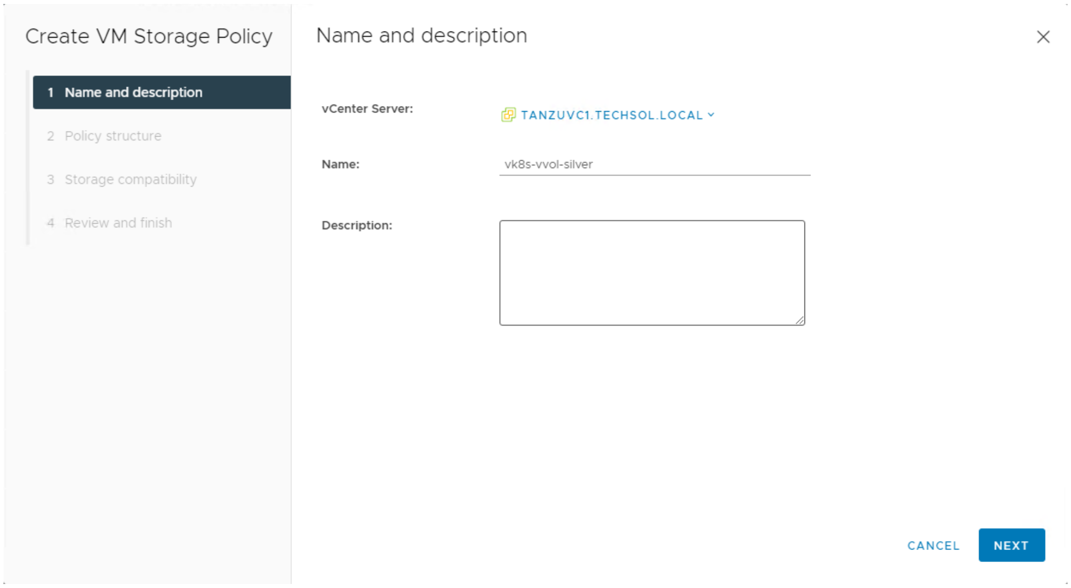


VM Storage Policies

CREATE

☐	Name	VC
☐	Host-local PMem Default Storage Policy	tanzuvc1.techsol.local
☐	Management Storage policy - Encrypti...	tanzuvc1.techsol.local

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



Create VM Storage Policy

1 Name and description

2 Policy structure

3 Storage compatibility

4 Review and finish

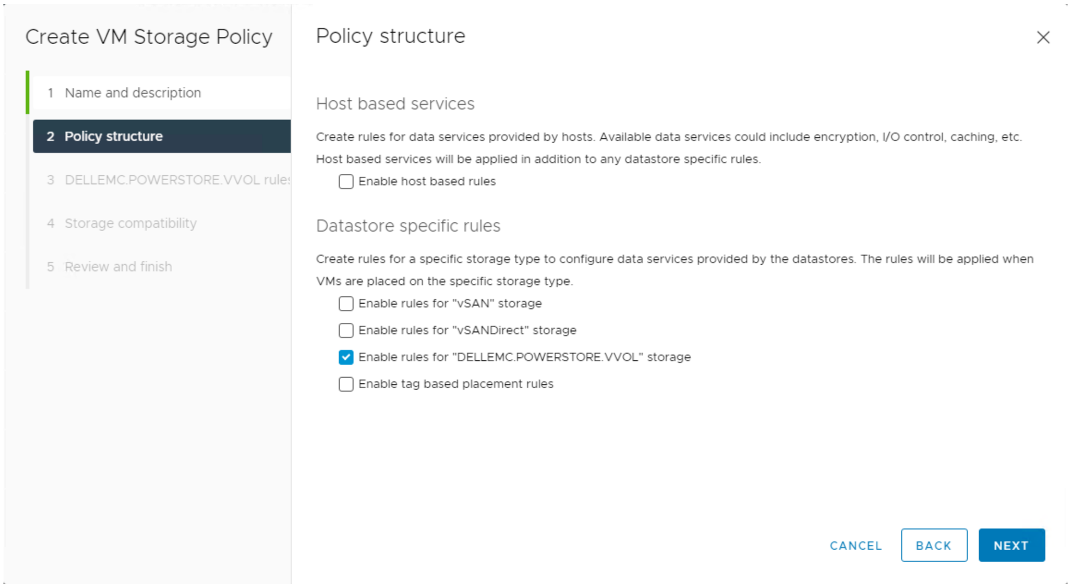
Name and description

vCenter Server: TANZUVC1.TECHSOL.LOCAL

Name: vk8s-vvol-silver

Description:

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



Create VM Storage Policy

1 Name and description

2 Policy structure

3 DELLEMC.POWERSTORE.VVOL 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.

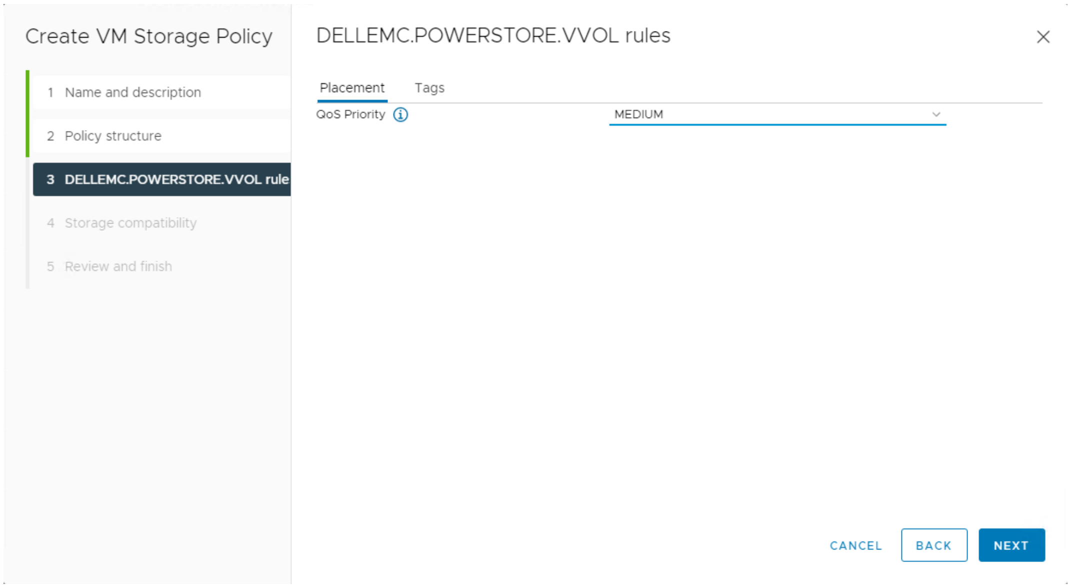
☐ Enable rules for "vSAN" storage

☐ Enable rules for "vSANDirect" storage

☑ Enable rules for "DELLEMC.POWERSTORE.VVOL" storage

☐ Enable tag based placement rules

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



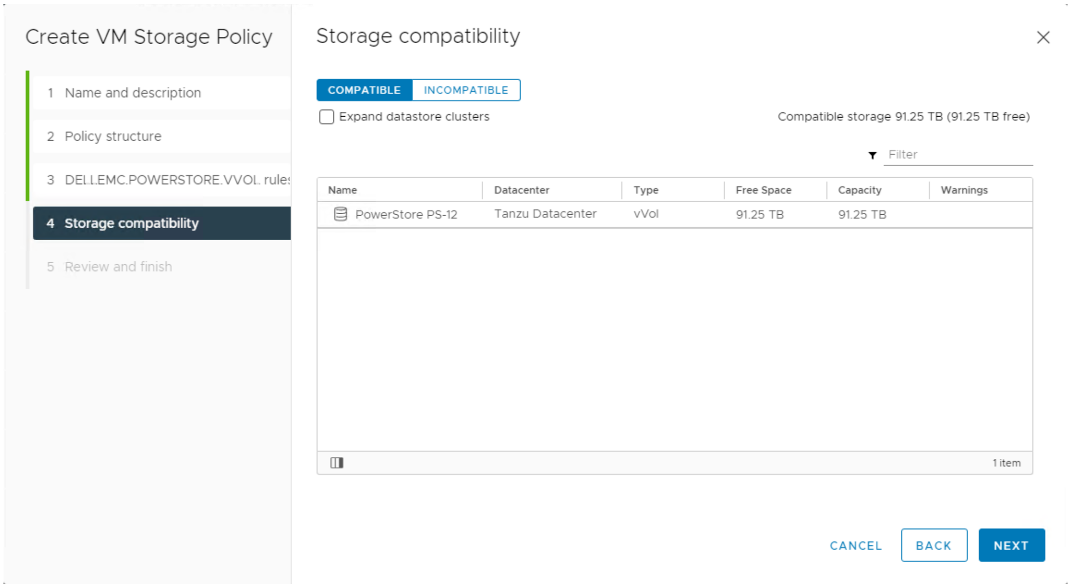
Create VM Storage Policy

1 Name and description
2 Policy structure
3 DELLEMC.POWERSTORE.VVOL rule
4 Storage compatibility
5 Review and finish

DELLEMC.POWERSTORE.VVOL rules

QoS Priority: MEDIUM

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 DELLEMC.POWERSTORE.VVOL rules
4 Storage compatibility
5 Review and finish

Storage compatibility

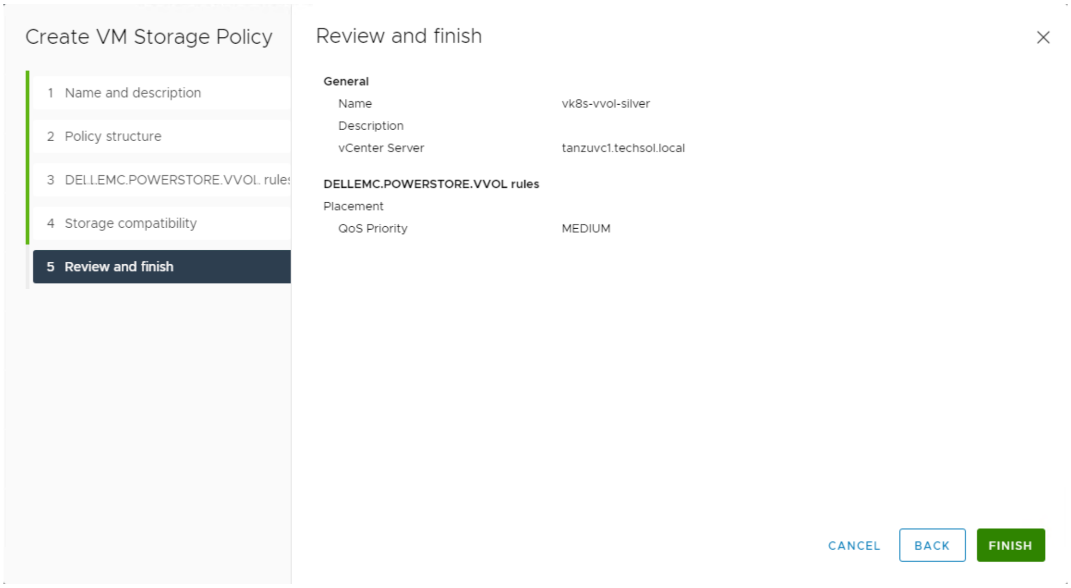
□ Expand datastore clusters

Compatible storage 91.25 TB (91.25 TB free)

Name	Datacenter	Type	Free Space	Capacity	Warnings
PowerStore PS-12	Tanzu Datacenter	vVol	91.25 TB	91.25 TB	

1 item

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



Review and finish

General

Name	vk8s-vvol-silver
Description	
vCenter Server	tanzuvc1.techsol.local

DELLEMC.POWERSTORE.VVOL rules

Placement

QoS Priority	MEDIUM

PowerStore X model specific configuration

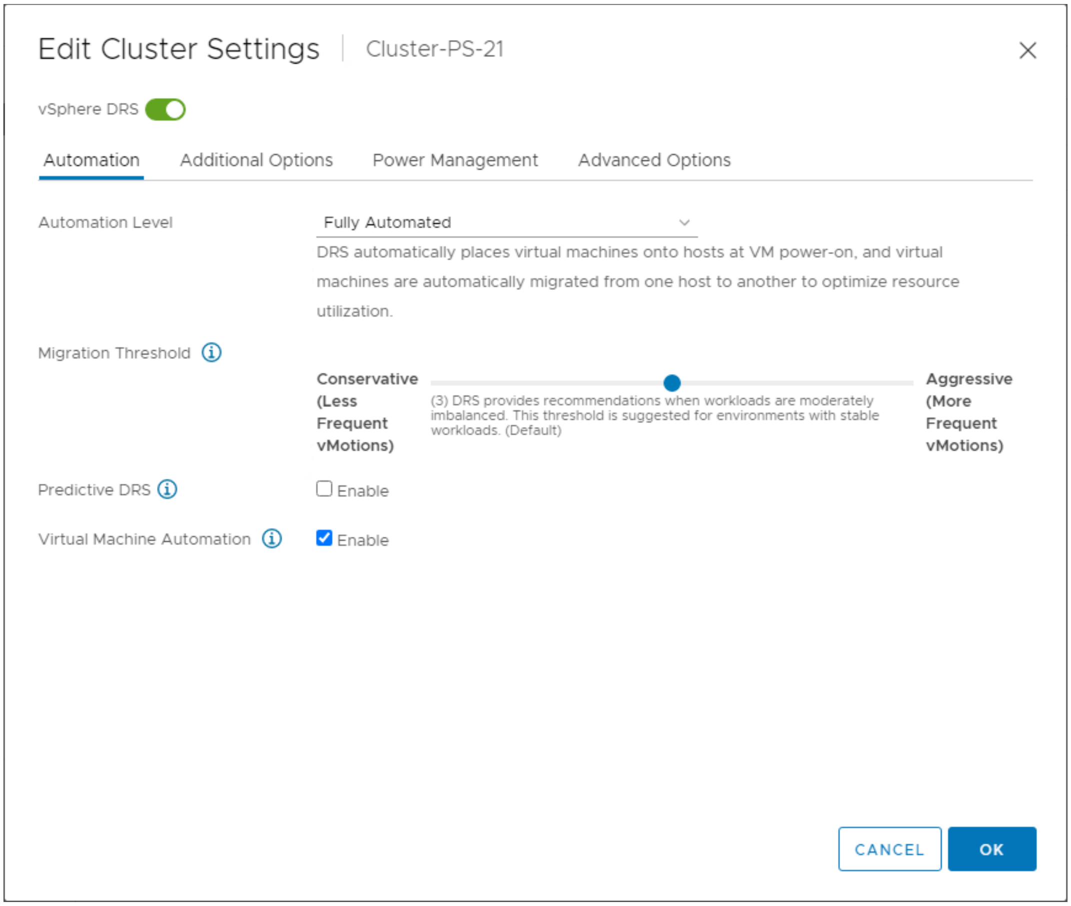
When deploying vSphere with Tanzu on a PowerStore X model, perform the following three steps before enabling Workload Management.

Configure a PowerStore X model cluster

A single PowerStore X model appliance contains two nodes which are deployed as two ESXi hosts. vSphere with Tanzu requires a minimum of three ESXi hosts. Therefore, a minimum of two PowerStore X appliances must be deployed in a PowerStore X model cluster. A two-appliance cluster would yield four nodes or four ESXi hosts, which meets the vSphere with Tanzu minimum requirement of three ESXi hosts

Configure DRS Automation Level

When the PowerStore X model is deployed, by default it sets the cluster DRS Automation Level to Partially Automated. vSphere with Tanzu requires that the cluster DRS Automation Level be set to Fully Automated. Modifying the DRS Automation Level is a feature that was introduced in PowerStore 2.1.



vSphere DRS ●

Automation Level: Fully Automated

DRS automatically places virtual machines onto hosts at VM power-on, and virtual machines are automatically migrated from one host to another to optimize resource utilization.

Migration Threshold: Conservative (Less Frequent vMotions) – Aggressive (More Frequent vMotions)

(3) DRS provides recommendations when workloads are moderately imbalanced. This threshold is suggested for environments with stable workloads. (Default)

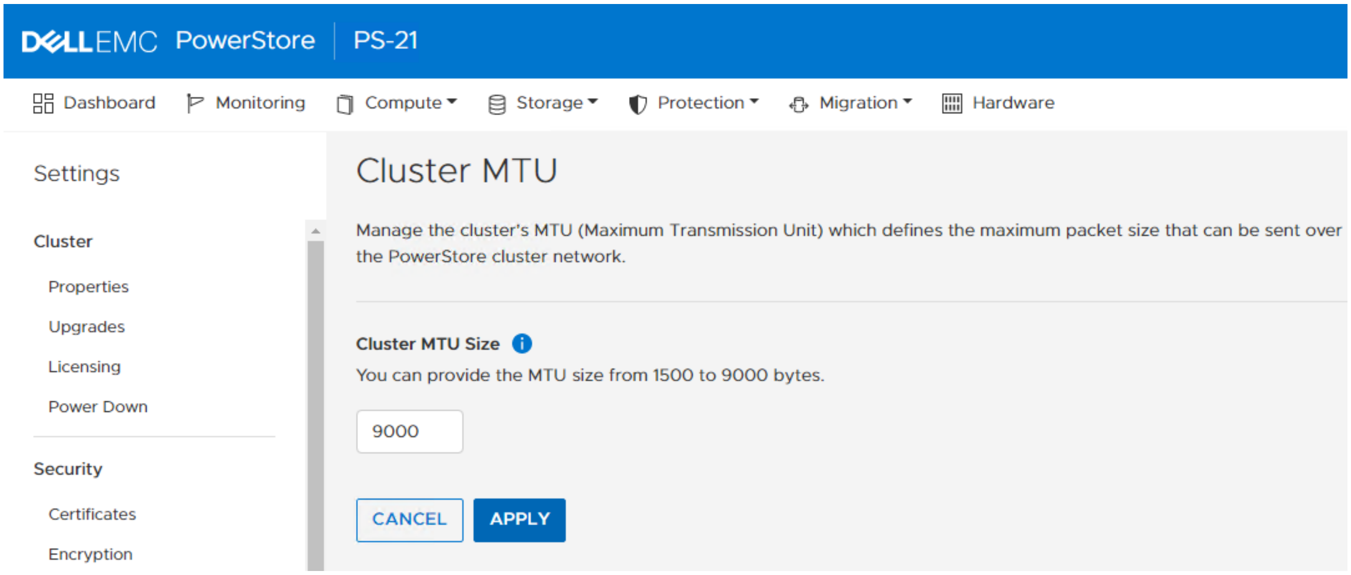
Predictive DRS: □ Enable

Virtual Machine Automation: ☑ Enable

Figure 3. Configure the DRS Automation Level for Fully Automated to meet vSphere with Tanzu requirements

Configure Cluster MTU

When the PowerStore X model is deployed, by default it sets the Cluster MTU to 1500 bytes. This Cluster MTU setting correlates to the vSphere Distributed Switch MTU which will also reflect the same value of 1500. vSphere with Tanzu requires jumbo frames support with a minimum MTU value of 1600. Dell Technologies recommends configuring the Cluster MTU in PowerStore Manager to the maximum value of 9000 bytes. Once completed, PowerStore Manager will automatically configure the Distributed Switch with the same MTU value of 9000 bytes.



Cluster MTU

Manage the cluster's MTU (Maximum Transmission Unit) which defines the maximum packet size that can be sent over the PowerStore cluster network.

Cluster MTU Size

You can provide the MTU size from 1500 to 9000 bytes.

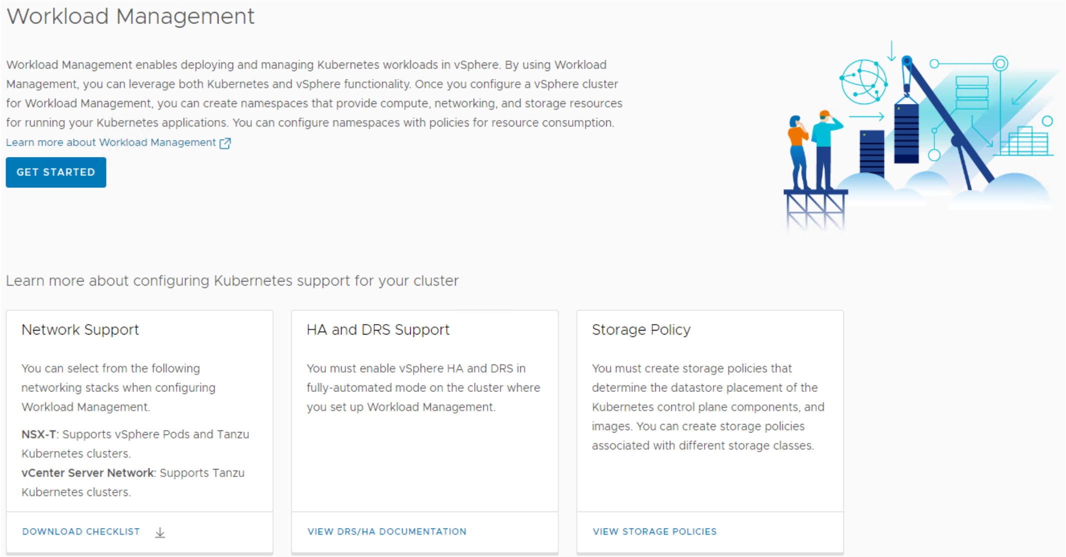
9000

Figure 4. Configure the Cluster MTU for 9000 Bytes to meet vSphere with Tanzu jumbo frames requirements

Note: Configuring MTU of the Distributed Switch outside of PowerStore Manager is not recommended and will result in Alerts being generated in PowerStore Manager.

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 will be shown as an example using NSX-T for networking and PowerStore for the storage. In the vSphere UI, use the pulldown menu to go to Workload Management. Click the Get Started button.


Workload Management

Workload Management enables deploying and managing Kubernetes workloads in vSphere. By using Workload Management, you can leverage both Kubernetes and vSphere functionality. Once you configure a vSphere cluster for Workload Management, you can create namespaces that provide compute, networking, and storage resources for running your Kubernetes applications. You can configure namespaces with policies for resource consumption.

Learn more about Workload Management

Learn more about configuring Kubernetes support for your cluster

Network Support

You can select from the following networking stacks when configuring Workload Management.

NSX-T: Supports vSphere Pods and Tanzu Kubernetes clusters.
vCenter Server Network: Supports Tanzu Kubernetes clusters.

HA and DRS Support

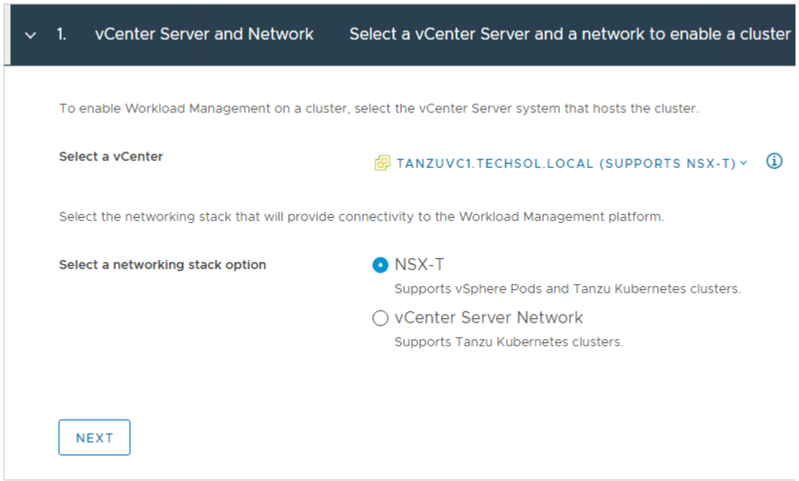
You must enable vSphere HA and DRS in fully-automated mode on the cluster where you set up Workload Management.

Storage Policy

You must create storage policies that determine the datastore placement of the Kubernetes control plane components, and images. You can create storage policies associated with different storage classes.

Figure 5. Enabling Workload Management to deploy vSphere with Tanzu

1. vCenter Server and Network: NSX-T will be used 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: TANZUVC1.TECHSOL.LOCAL (SUPPORTS NSX-T)

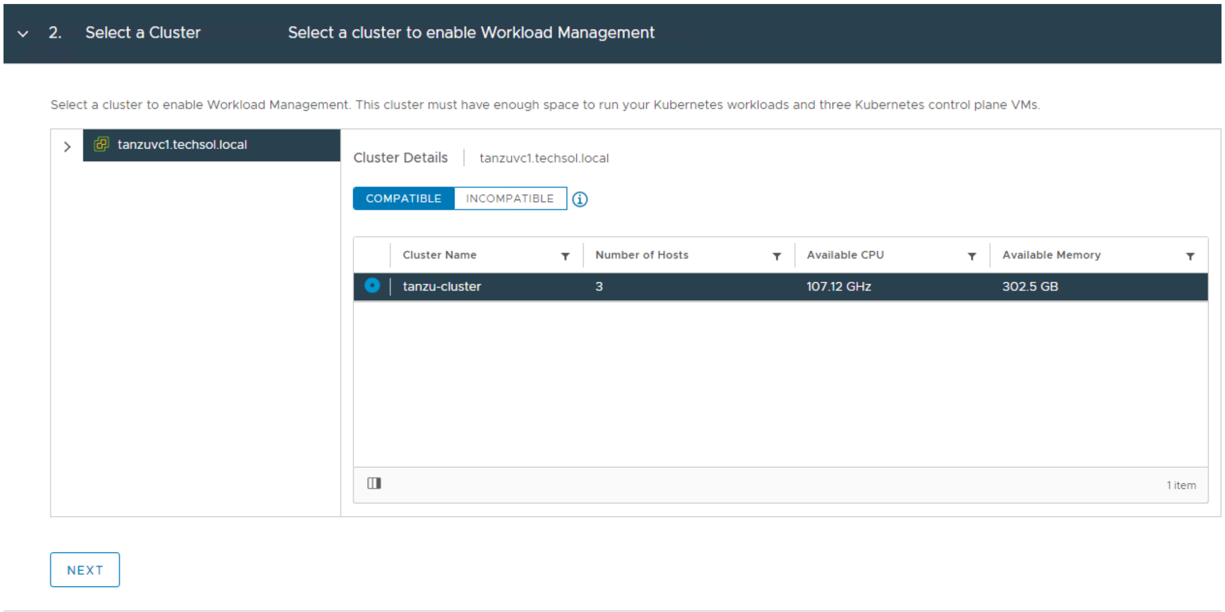
Select the networking stack that will provide connectivity to the Workload Management platform.

Select a networking stack option:

● NSX-T
Supports vSphere Pods and Tanzu Kubernetes clusters.

○ vCenter Server Network
Supports Tanzu Kubernetes clusters.

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



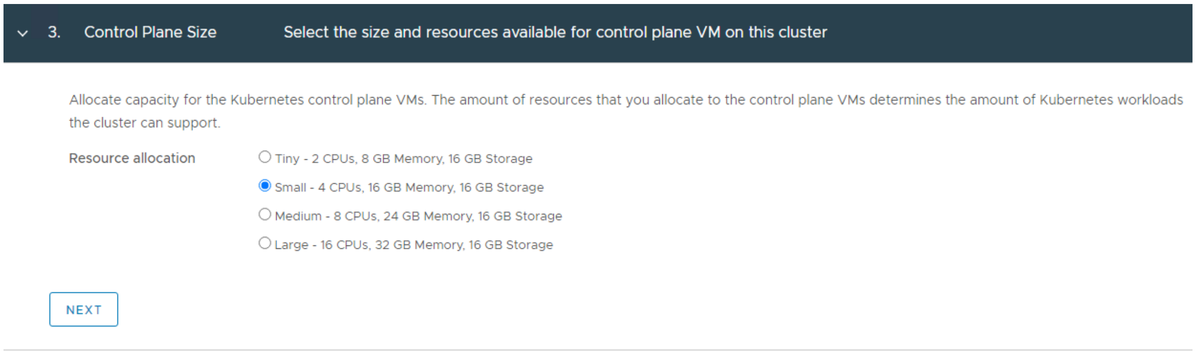
2. Select a Cluster — Select a cluster to enable Workload Management

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

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

1 item

3. Choose a Control Plane Size.



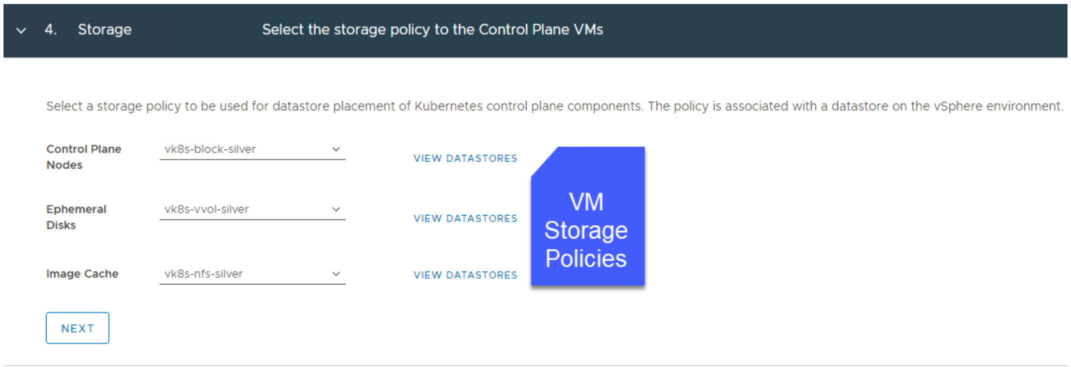
3. Control Plane Size — Select the size and resources available for control plane VM on this cluster

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

4. Choose 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

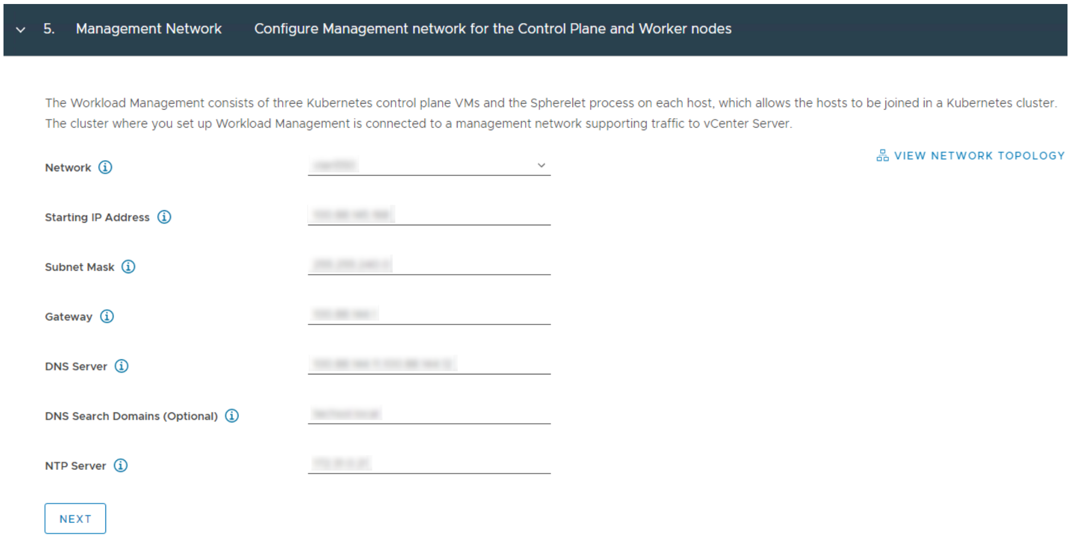
Select a storage policy to be used for datastore placement of Kubernetes control plane components. The policy is associated with a datastore on the vSphere environment.

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

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.

5. Provide details for the Management Network.

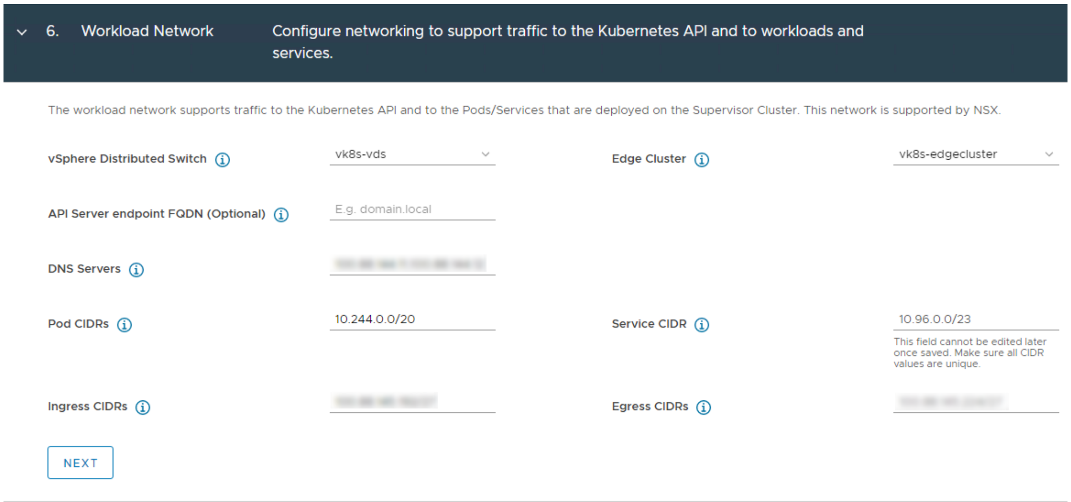


5. Management Network — Configure Management network for the Control Plane and Worker nodes

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.

Network ⓘ	[redacted]
Starting IP Address ⓘ	[redacted]
Subnet Mask ⓘ	[redacted]
Gateway ⓘ	[redacted]
DNS Server ⓘ	[redacted]
DNS Search Domains (Optional) ⓘ	[redacted]
NTP Server ⓘ	[redacted]

6. 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: vk8s-vds

Edge Cluster: vk8s-edgecluster

API Server endpoint FQDN (Optional): E.g. domain.local

DNS Servers: [redacted]

Pod CIDRs: 10.244.0.0/20

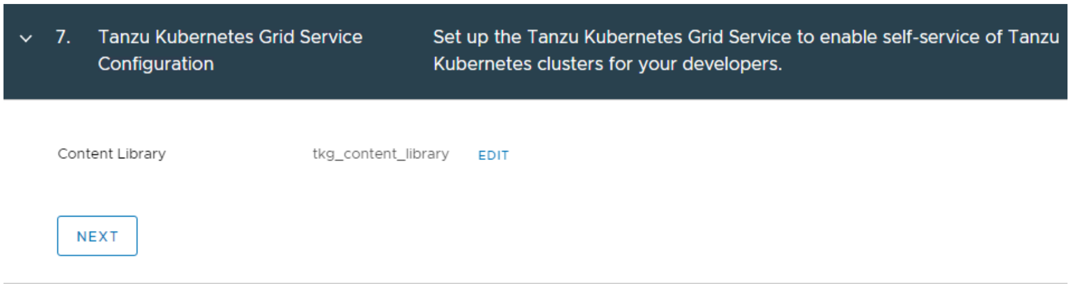
Service CIDR: 10.96.0.0/23

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

Ingress CIDRs: [redacted]

Egress CIDRs: [redacted]

7. 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.

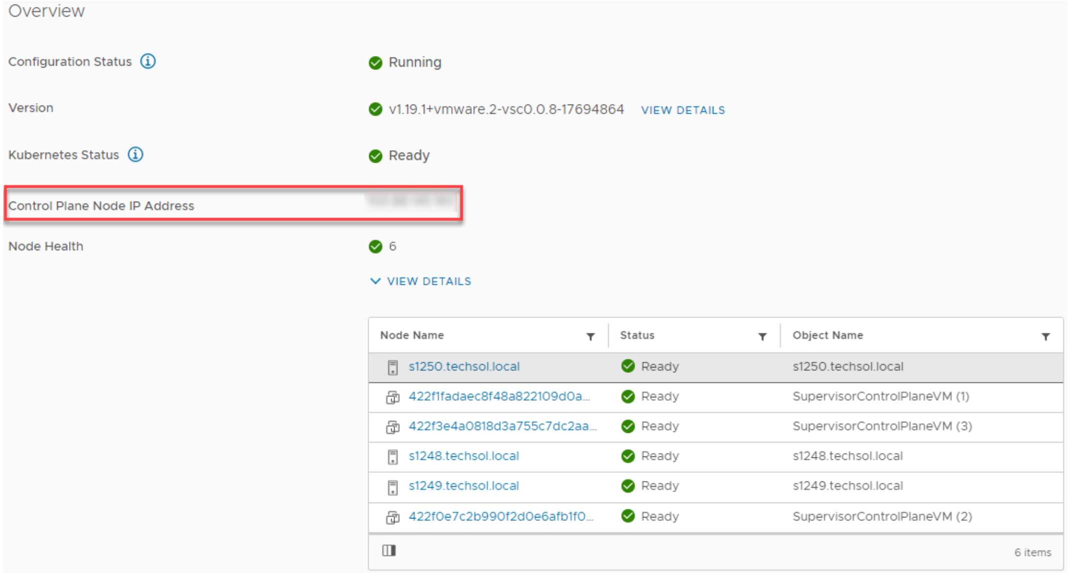


7. Tanzu Kubernetes Grid Service Configuration — Set up the Tanzu Kubernetes Grid Service to enable self-service of Tanzu Kubernetes clusters for your developers.

Content Library: tkg_content_library

8. 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 which **kubectl** commands will be issued against.



Overview

Configuration Status ✓ Running

Version ✓ v1.19.1+vmware.2-vsc0.0.8-17694864 VIEW DETAILS

Kubernetes Status ✓ Ready

Control Plane Node IP Address [redacted]

Node Health ✓ 6

VIEW DETAILS

Node Name	Status	Object Name
s1250.techsol.local	✓ Ready	s1250.techsol.local
422f1fadaec8f48a822109d0a...	✓ Ready	SupervisorControlPlaneVM (1)
422f3e4a0818d3a755c7dc2aa...	✓ Ready	SupervisorControlPlaneVM (3)
s1248.techsol.local	✓ Ready	s1248.techsol.local
s1249.techsol.local	✓ Ready	s1249.techsol.local
422f0e7c2b990f2d0e6afb1f0...	✓ Ready	SupervisorControlPlaneVM (2)

6 items

Figure 6. vSphere with Tanzu Supervisor Cluster Overview and Control Plane Node IP Address

Day 2 Kubernetes

Introduction

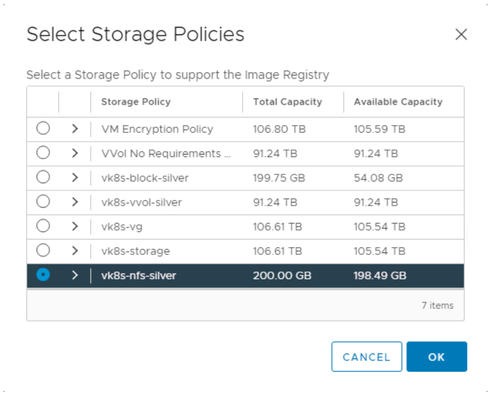
With design and deployment (sometimes referred to as Day 0 and Day 1) complete, we can now start to focus on some of the operational aspects of vSphere with Tanzu and Tanzu Kubernetes Clusters along with their relationship to PowerStore storage. The following sections will explore these areas.

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 on 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** will be used.



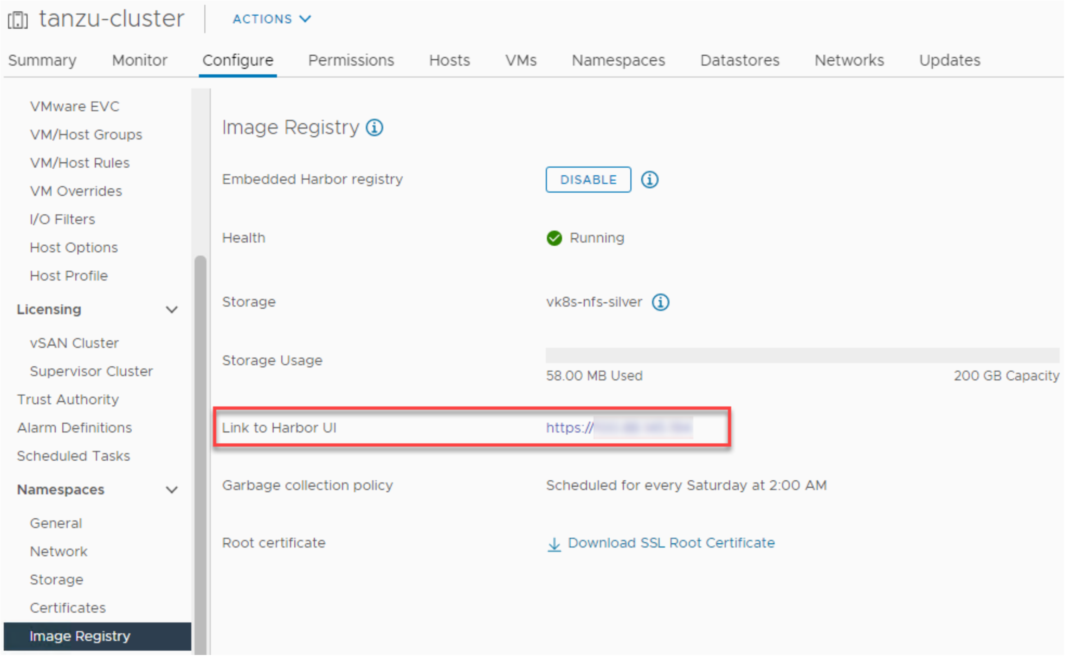
Select Storage Policies

Select a Storage Policy to support the Image Registry

	Storage Policy	Total Capacity	Available Capacity
○	VM Encryption Policy	106.80 TB	105.59 TB
○	VVol No Requirements ...	91.24 TB	91.24 TB
○	vk8s-block-silver	199.75 GB	54.08 GB
○	vk8s-vvol-silver	91.24 TB	91.24 TB
○	vk8s-vg	106.61 TB	105.54 TB
○	vk8s-storage	106.61 TB	105.54 TB
●	vk8s-nfs-silver	200.00 GB	198.49 GB

7 items

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.



tanzu-cluster | ACTIONS

Summary | Monitor | Configure | Permissions | Hosts | VMs | Namespaces | Datastores | Networks | Updates

- VMware EVC
- VM/Host Groups
- VM/Host Rules
- VM Overrides
- I/O Filters
- Host Options
- Host Profile
- Licensing
  - vSAN Cluster
  - Supervisor Cluster
- Trust Authority
- Alarm Definitions
- Scheduled Tasks
- Namespaces
  - General
  - Network
  - Storage
  - Certificates
  - Image Registry

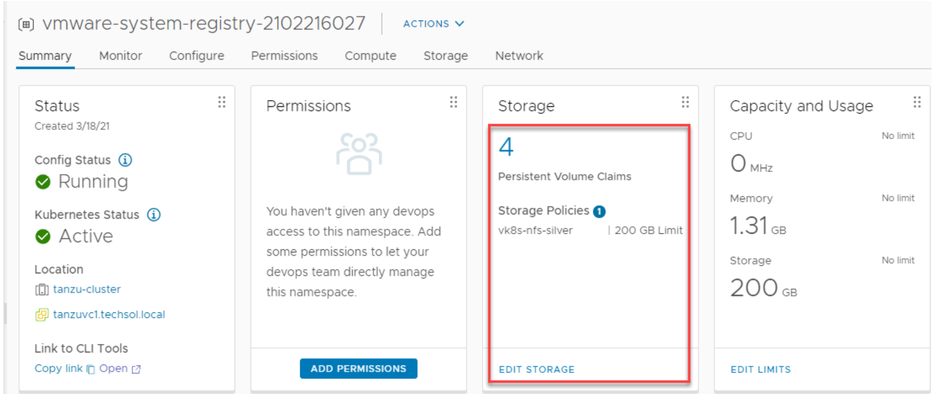
Image Registry

Embedded Harbor registry	DISABLE
Health	Running
Storage	vk8s-nfs-silver
Storage Usage	58.00 MB Used 200 GB Capacity
Link to Harbor UI	https://[redacted]
Garbage collection policy	Scheduled for every Saturday at 2:00 AM
Root certificate	Download SSL Root Certificate

Figure 7. 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.

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



vmware-system-registry-2102216027 | ACTIONS

Summary | Monitor | Configure | Permissions | Compute | Storage | Network

Status
Created 3/18/21
Config Status
Running
Kubernetes Status
Active
Location
tanzu-cluster
tanzuvc1.techsol.local
Link to CLI Tools
Copy link Open

Permissions
You haven't given any devops access to this namespace. Add some permissions to let your devops team directly manage this namespace.
ADD PERMISSIONS

Storage
4
Persistent Volume Claims
Storage Policies 1
vk8s-nfs-silver | 200 GB Limit
EDIT STORAGE

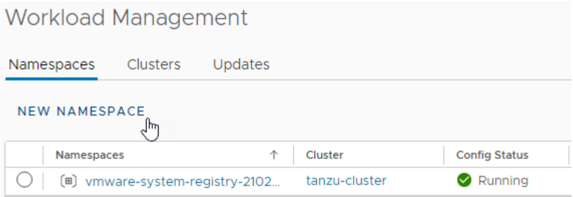
Capacity and Usage
CPU No limit
0 MHz
Memory No limit
1.31 GB
Storage No limit
200 GB
EDIT LIMITS

Figure 8. Summary page of the Harbor application's Namespace

Creating a vSphere Namespace

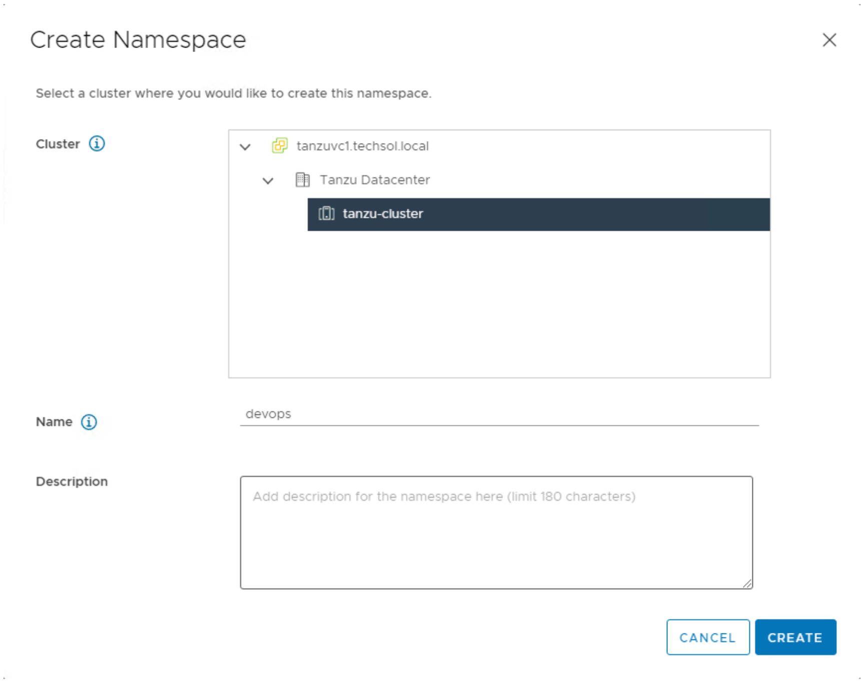
Containerized applications are deployed across one or more vSphere Pods in a vSphere Namespace. A vSphere Namespace is backed by a Resource Pool which is used to control resource limits. In the following example, a new vSphere Namespace will be created to explore storage integration further. To create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

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



Namespaces	Cluster	Config Status
○ vmware-system-registry-2102...	tanzu-cluster	✓ Running

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



Select a cluster where you would like to create this namespace.

Cluster: tanzuvc1.techsol.local > Tanzu Datacenter > tanzu-cluster

Name: devops

Description: Add description for the namespace here (limit 180 characters)

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.

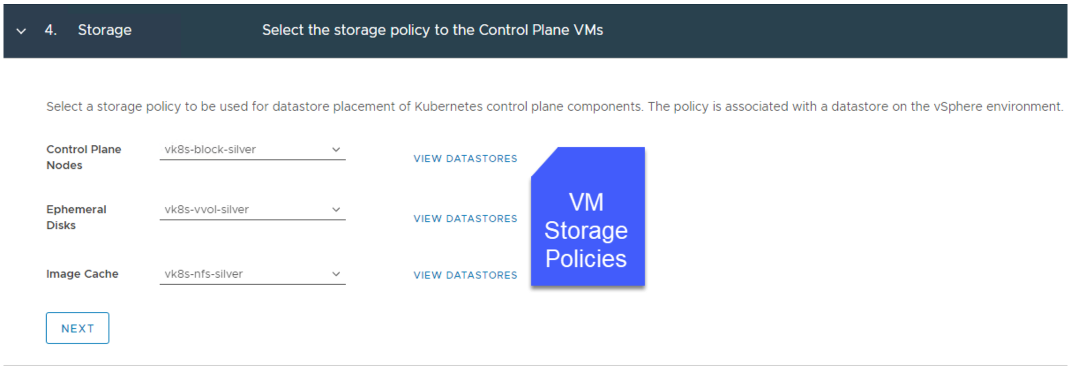
The devops vSphere Namespace is now created. The next step would typically be to delegate permissions to a DevOps team. These steps are covered in depth in the VMware document [vSphere with Tanzu Configuration and Management.]

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.



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.

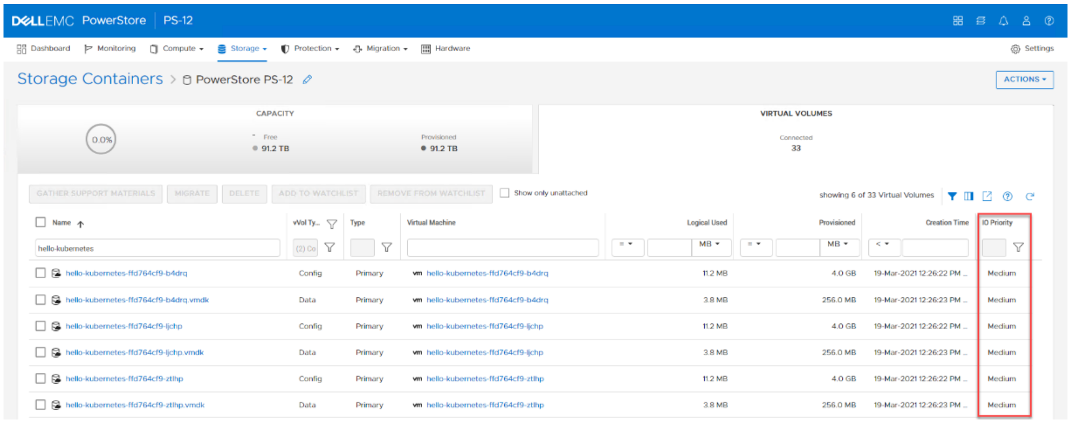
Control Plane Nodes: vk8s-block-silver

Ephemeral Disks: vk8s-vvol-silver

Image Cache: vk8s-nfs-silver

Figure 9. 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.



CAPACITY

0.0%

Free: 91.2 TB

Provisioned: 91.2 TB

VIRTUAL VOLUMES

Connected: 33

□ Show only unattached

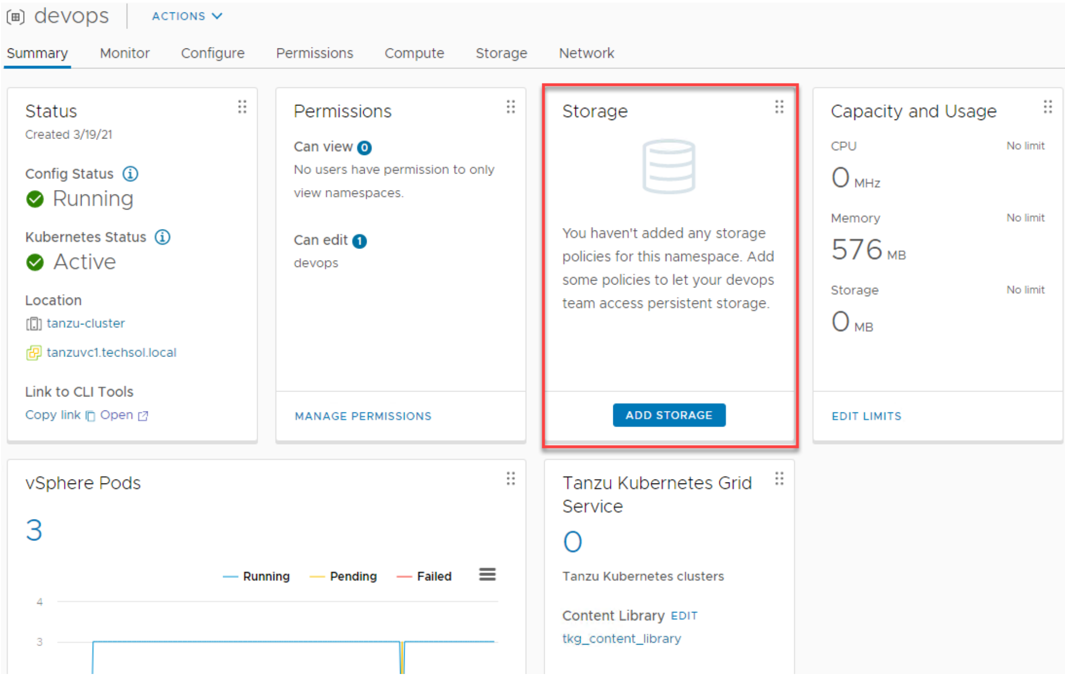
showing 6 of 33 Virtual Volumes

Name	vVol Ty...	Type	Virtual Machine	Logical Used	Provisioned	Creation Time	IO Priority
hello-kubernetes	(2) Co			MB	MB		
hello-kubernetes-ffd764cf9-b4drq	Config	Primary	hello-kubernetes-ffd764cf9-b4drq	11.2 MB	4.0 GB	19-Mar-2021 12:26:22 PM ...	Medium
hello-kubernetes-ffd764cf9-b4drq.vmdk	Data	Primary	hello-kubernetes-ffd764cf9-b4drq	3.8 MB	256.0 MB	19-Mar-2021 12:26:23 PM ...	Medium
hello-kubernetes-ffd764cf9-ljchp	Config	Primary	hello-kubernetes-ffd764cf9-ljchp	11.2 MB	4.0 GB	19-Mar-2021 12:26:22 PM ...	Medium
hello-kubernetes-ffd764cf9-ljchp.vmdk	Data	Primary	hello-kubernetes-ffd764cf9-ljchp	3.8 MB	256.0 MB	19-Mar-2021 12:26:23 PM ...	Medium
hello-kubernetes-ffd764cf9-ztlhp	Config	Primary	hello-kubernetes-ffd764cf9-ztlhp	11.2 MB	4.0 GB	19-Mar-2021 12:26:22 PM ...	Medium
hello-kubernetes-ffd764cf9-ztlhp.vmdk	Data	Primary	hello-kubernetes-ffd764cf9-ztlhp	3.8 MB	256.0 MB	19-Mar-2021 12:26:23 PM ...	Medium

Figure 10. 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.



Status
Created 3/19/21

Config Status
Running

Kubernetes Status
Active

Location
tanzu-cluster
tanzuvc1.techsol.local

Link to CLI Tools
Copy link Open

Permissions

Can view 0
No users have permission to only view namespaces.

Can edit 1
devops

Storage

You haven't added any storage policies for this namespace. Add some policies to let your devops team access persistent storage.

Capacity and Usage

CPU: 0 MHz (No limit)

Memory: 576 MB (No limit)

Storage: 0 MB (No limit)

vSphere Pods
3

Tanzu Kubernetes Grid Service
0
Tanzu Kubernetes clusters

Content Library EDIT
tkg_content_library

Figure 11. Devops vSphere Namespace showing no storage assignment

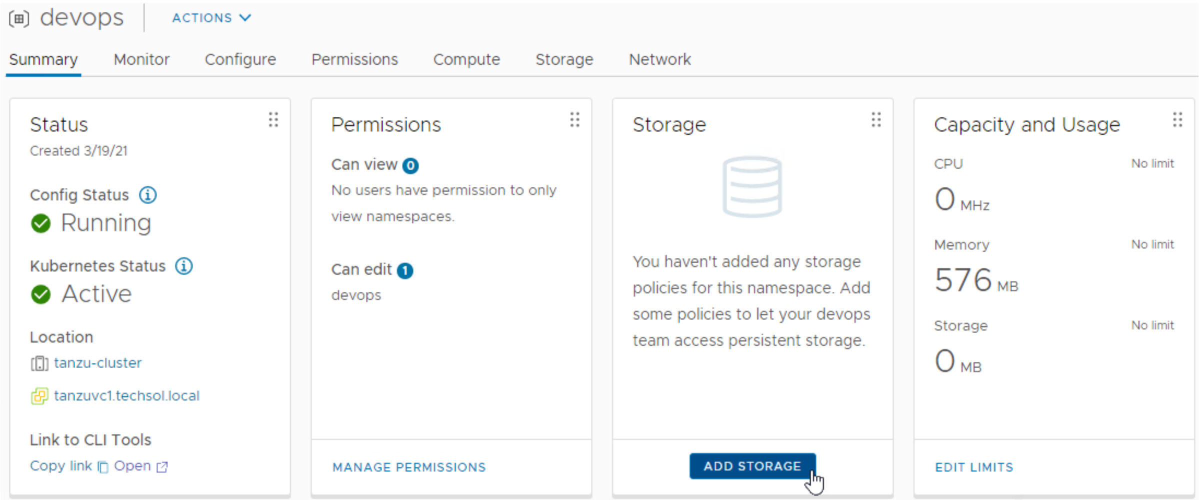
Adding, limiting, and removing storage in a vSphere Namespace

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. Drill into the vSphere Namespace requiring persistent storage. Click Add Storage.



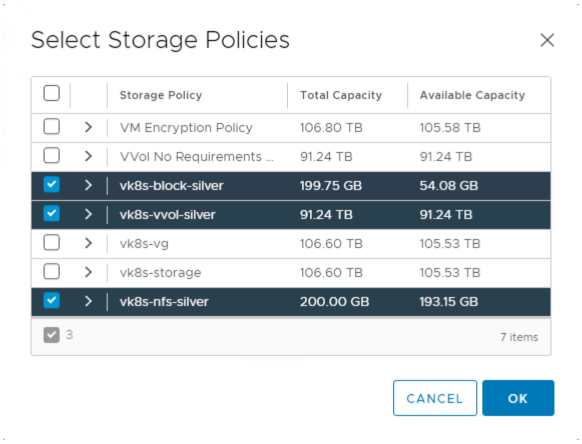
Status
Created 3/19/21
Config Status: Running
Kubernetes Status: Active
Location: tanzu-cluster, tanzuvc1.techsol.local
Link to CLI Tools: Copy link, Open

Permissions
Can view 0
No users have permission to only view namespaces.
Can edit 1
devops

Storage
You haven't added any storage policies for this namespace. Add some policies to let your devops team access persistent storage.

Capacity and Usage
CPU: 0 MHz (No limit)
Memory: 576 MB (No limit)
Storage: 0 MB (No limit)

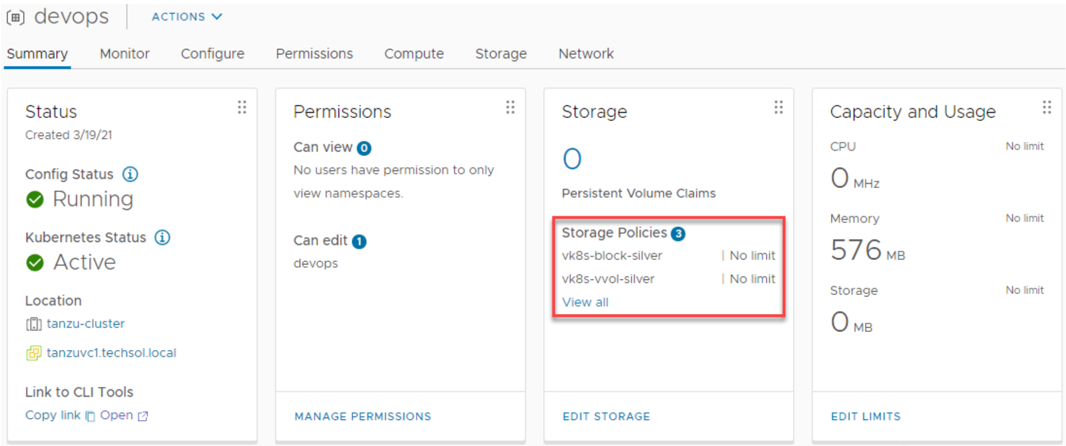
2. Select one or more Storage Policies to be used to back persistent storage. When a VM Storage Policy is created and assigned for use in vSphere with Tanzu, vSphere with Tanzu creates a matching Kubernetes storage class in the Supervisor Namespace. These storage classes will also be replicated to any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



☐	Storage Policy	Total Capacity	Available Capacity
☐	VM Encryption Policy	106.80 TB	105.58 TB
☐	VVol No Requirements ...	91.24 TB	91.24 TB
☑	vk8s-block-silver	199.75 GB	54.08 GB
☑	vk8s-vvol-silver	91.24 TB	91.24 TB
☐	vk8s-vg	106.60 TB	105.53 TB
☐	vk8s-storage	106.60 TB	105.53 TB
☑	vk8s-nfs-silver	200.00 GB	193.15 GB

☑ 3 — 7 items

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



Status
Created 3/19/21
Config Status
Running
Kubernetes Status
Active
Location
tanzu-cluster
tanzuvc1.techsol.local
Link to CLI Tools

Permissions
Can view 0
No users have permission to only view namespaces.
Can edit 1
devops

Storage
0
Persistent Volume Claims
Storage Policies 3
vk8s-block-silver | No limit
vk8s-vvol-silver | No limit
View all

Capacity and Usage
CPU No limit
0 MHz
Memory No limit
576 MB
Storage No limit
0 MB

Figure 12. 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.

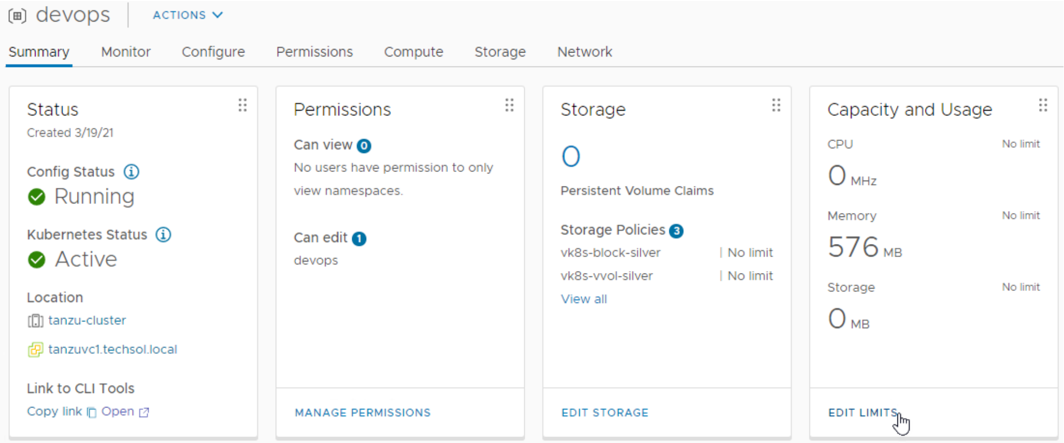
```
C:\>kubectl describe storageclass
Name:                  vk8s-block-silver
IsDefaultClass:        No
Annotations:
cns.vmware.com/StoragePoolTypeHint=cns.vmware.com/VMFS
Provisioner:           csi.vsphere.vmware.com
Parameters:            storagePolicyID=475fdecf-f339-4f99-9013-
87edd1ab6cdb
AllowVolumeExpansion:  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 as 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. Drill into to the vSphere Namespace. Under Capacity and Usage, click Edit Limits.



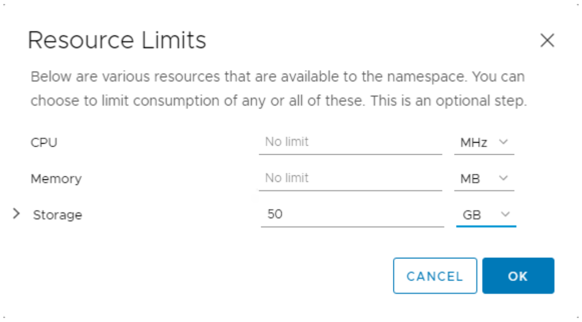
Status
Created 3/19/21
Config Status: Running
Kubernetes Status: Active
Location: tanzu-cluster, tanzuvc1.techsol.local
Link to CLI Tools: Copy link, Open

Permissions
Can view 0: No users have permission to only view namespaces.
Can edit 1: devops

Storage
0 Persistent Volume Claims
Storage Policies 3
vk8s-block-silver | No limit
vk8s-vvol-silver | No limit
View all

Capacity and Usage
CPU No limit: 0 MHz
Memory No limit: 576 MB
Storage No limit: 0 MB

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

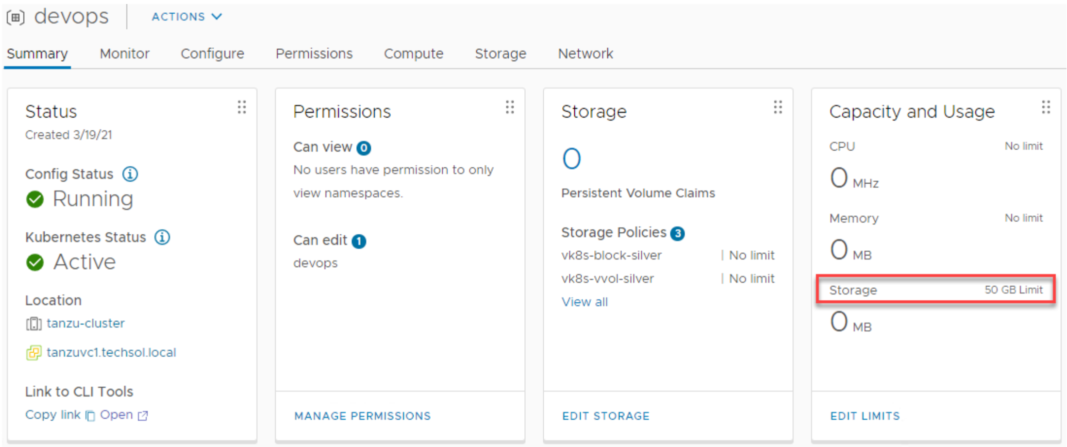


Resource Limits

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

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



Status
Created 3/19/21
Config Status: Running
Kubernetes Status: Active
Location: tanzu-cluster, tanzuvc1.techsol.local
Link to CLI Tools: Copy link, Open

Permissions
Can view 0: No users have permission to only view namespaces.
Can edit 1: devops

Storage
0 Persistent Volume Claims
Storage Policies 3
vk8s-block-silver | No limit
vk8s-vvol-silver | No limit
View all

Capacity and Usage
CPU No limit: 0 MHz
Memory No limit: 0 MB
Storage 50 GB Limit: 0 MB

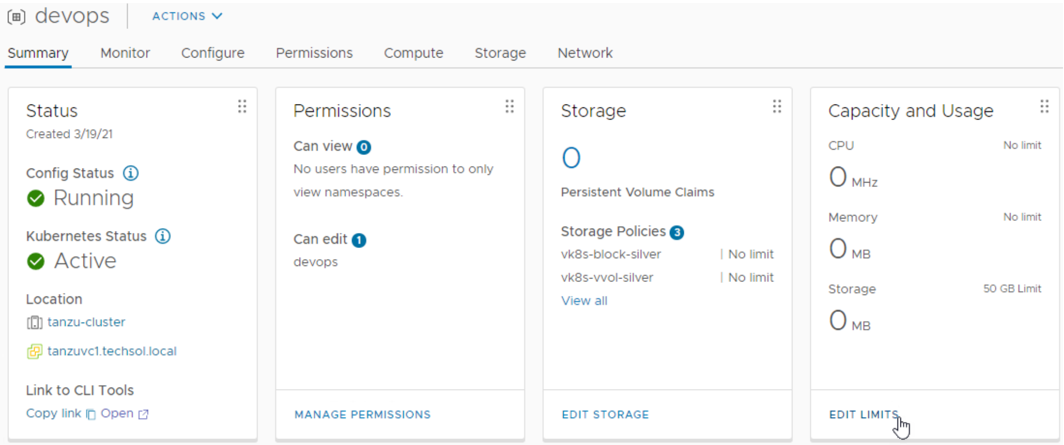
Figure 13. Viewing a global storage limit in a vSphere Namespace

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. Drill into to the vSphere Namespace. Under Capacity and Usage, click Edit Limits.



Status
Created 3/19/21
Config Status: Running
Kubernetes Status: Active
Location: tanzu-cluster, tanzuvc1.techsol.local
Link to CLI Tools: Copy link, Open

Permissions
Can view 0
No users have permission to only view namespaces.
Can edit 1
devops

Storage
0 Persistent Volume Claims
Storage Policies 3

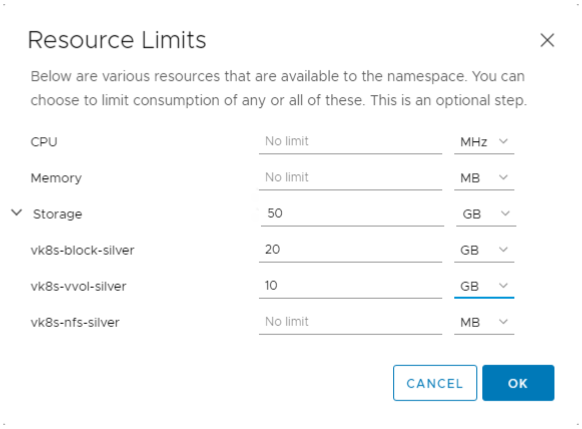
vk8s-block-silver	No limit
vk8s-vvol-silver	No limit

View all

Capacity and Usage

CPU	0 MHz	No limit
Memory	0 MB	No limit
Storage	0 MB	50 GB Limit

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

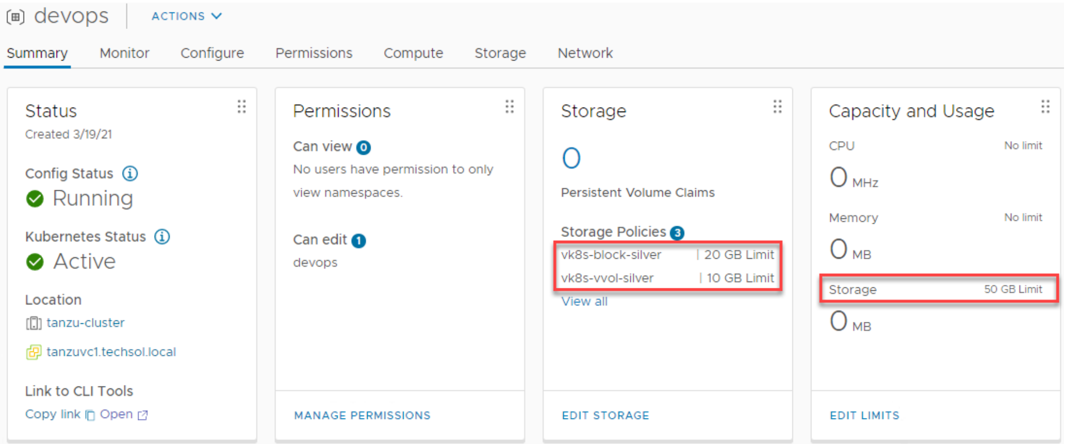


Resource Limits

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
vk8s-block-silver	20	GB
vk8s-vvol-silver	10	GB
vk8s-nfs-silver	No limit	MB

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.



Status
Created 3/19/21
Config Status: Running
Kubernetes Status: Active
Location: tanzu-cluster, tanzuvc1.techsol.local
Link to CLI Tools: Copy link, Open

Permissions
Can view 0
No users have permission to only view namespaces.
Can edit 1
devops

Storage
0 Persistent Volume Claims
Storage Policies 3

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

View all

Capacity and Usage

CPU	0 MHz	No limit
Memory	0 MB	No limit
Storage	0 MB	50 GB Limit

Figure 14. Viewing individual per storage limits in a vSphere Namespace

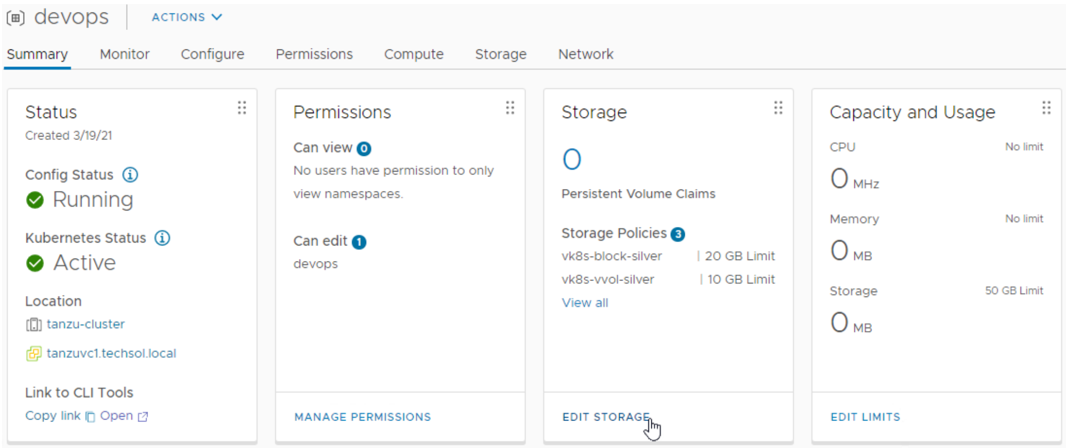
Using the **kubectl describe namespace** command, the same 50 GB global storage limit is visible as well as the 20 GB and 10 GB individual per storage limits for block and vVol respectively.

```
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. Drill into to the vSphere Namespace. Click Edit Storage.



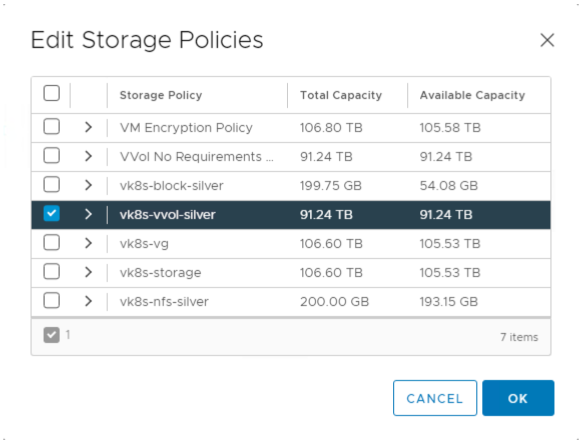
Status
Created 3/19/21
Config Status
Running
Kubernetes Status
Active
Location
tanzu-cluster
tanzuvc1.techsol.local
Link to CLI Tools
Copy link Open

Permissions
Can view 0
No users have permission to only view namespaces.
Can edit 1
devops
MANAGE PERMISSIONS

Storage
0
Persistent Volume Claims
Storage Policies 3
vk8s-block-silver | 20 GB Limit
vk8s-vvol-silver | 10 GB Limit
View all
EDIT STORAGE

Capacity and Usage
CPU No limit
0 MHz
Memory No limit
0 MB
Storage 50 GB Limit
0 MB
EDIT LIMITS

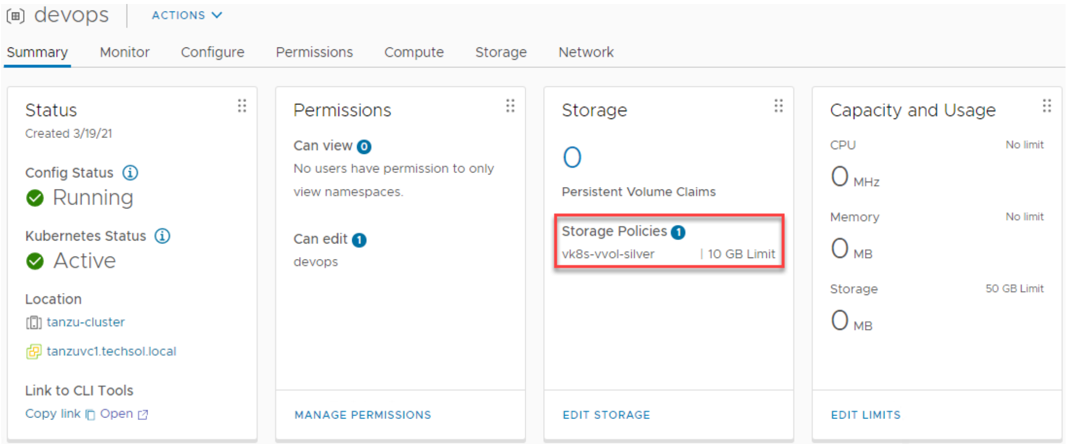
2. Clear the storage to be removed. Storage that should remain should be checked. Click OK.


Edit Storage Policies

☐	Storage Policy	Total Capacity	Available Capacity
☐	VM Encryption Policy	106.80 TB	105.58 TB
☐	VVol No Requirements ...	91.24 TB	91.24 TB
☐	vk8s-block-silver	199.75 GB	54.08 GB
☑	vk8s-vvol-silver	91.24 TB	91.24 TB
☐	vk8s-vg	106.60 TB	105.53 TB
☐	vk8s-storage	106.60 TB	105.53 TB
☐	vk8s-nfs-silver	200.00 GB	193.15 GB

☑ 1 7 items

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.



Status
Created 3/19/21
Config Status
Running
Kubernetes Status
Active
Location
tanzu-cluster
tanzuvc1.techsol.local
Link to CLI Tools
Copy link Open

Permissions
Can view 0
No users have permission to only view namespaces.
Can edit 1
devops
MANAGE PERMISSIONS

Storage
0
Persistent Volume Claims
Storage Policies 1
vk8s-vvol-silver | 10 GB Limit
EDIT STORAGE

Capacity and Usage
CPU No limit
0 MHz
Memory No limit
0 MB
Storage 50 GB Limit
0 MB
EDIT LIMITS

Figure 15. 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:

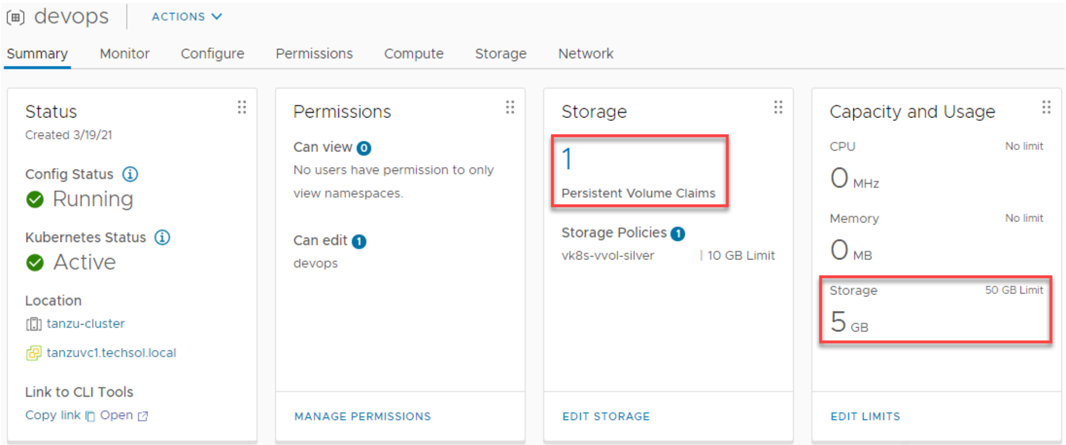
1. Create the persistent volume claim in a text editor and save as a .yaml file. In this example, 5 GB is being requested from the **vk8s-vvol-silver** storage class.

```
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.



Status
Created 3/19/21
Config Status: Running
Kubernetes Status: Active
Location: tanzu-cluster, tanzuvc1.techsol.local
Link to CLI Tools: Copy link, Open

Permissions
Can view 0: No users have permission to only view namespaces.
Can edit 1: devops

Storage
1 Persistent Volume Claims
Storage Policies 1: vk8s-vvol-silver | 10 GB Limit

Capacity and Usage
CPU: 0 MHz (No limit)
Memory: 0 MB (No limit)
Storage: 5 GB (50 GB Limit)

Figure 16. 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.

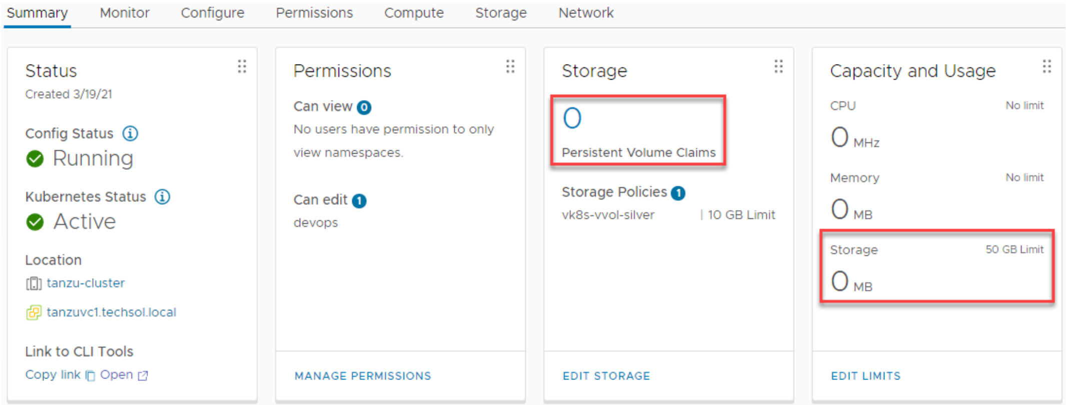
To delete a dynamic persistent volume claim, follow these steps:

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.



Status
Created 3/19/21
Config Status: Running
Kubernetes Status: Active
Location: tanzu-cluster, tanzuvc1.techsol.local
Link to CLI Tools

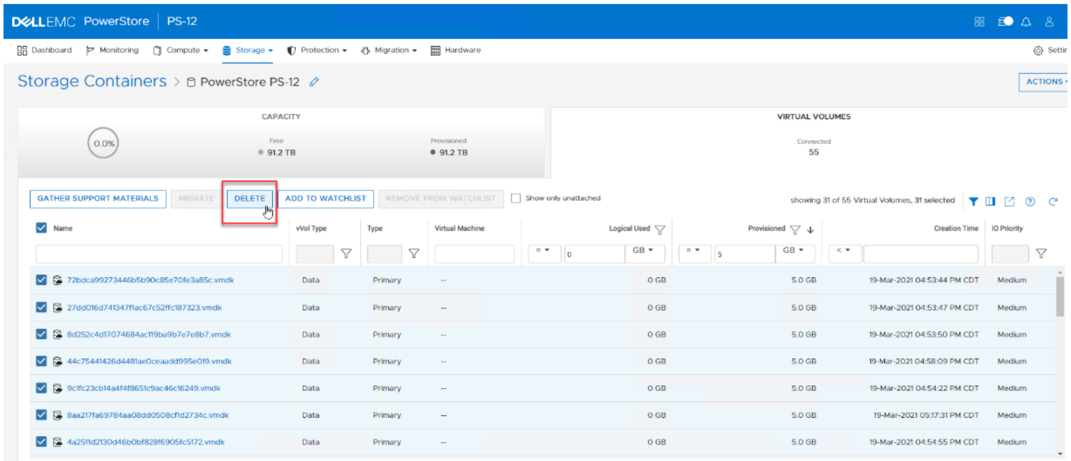
Permissions
Can view 0: No users have permission to only view namespaces.
Can edit 1: devops

Storage
0 Persistent Volume Claims
Storage Policies 1: vk8s-vvol-silver | 10 GB Limit

Capacity and Usage
CPU: 0 MHz (No limit)
Memory: 0 MB (No limit)
Storage: 0 MB (50 GB Limit)

Figure 17. 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.



Storage Containers > PowerStore PS-12

CAPACITY: 0.0% — Free 91.2 TB — Provisioned 91.2 TB

VIRTUAL VOLUMES: Connected 55

□ Show only unattached

showing 31 of 55 Virtual Volumes, 31 selected

Name	vVol Type	Type	Virtual Machine	Logical Used	Provisioned	Creation Time	IO Priority
☑ 72bdca99273446b5b90c85e70fe3a85c.vmdk	Data	Primary	--	0 GB	5.0 GB	19-Mar-2021 04:53:44 PM CDT	Medium
☑ 27dd016d741347f1ac67c52ffc187323.vmdk	Data	Primary	--	0 GB	5.0 GB	19-Mar-2021 04:53:47 PM CDT	Medium
☑ 8d252c4d17074684ac119ba9b7e7e8b7.vmdk	Data	Primary	--	0 GB	5.0 GB	19-Mar-2021 04:53:50 PM CDT	Medium
☑ 44c75441426d4481ae0ceaadd995e019.vmdk	Data	Primary	--	0 GB	5.0 GB	19-Mar-2021 04:58:09 PM CDT	Medium
☑ 9c1fc23cb14a4f4f8651c9ac46c16249.vmdk	Data	Primary	--	0 GB	5.0 GB	19-Mar-2021 04:54:22 PM CDT	Medium
☑ 8aa217fa69784aa08dd0508cf1d2734c.vmdk	Data	Primary	--	0 GB	5.0 GB	19-Mar-2021 05:17:31 PM CDT	Medium
☑ 4a2511d2130d46b0bf828f6905fc5172.vmdk	Data	Primary	--	0 GB	5.0 GB	19-Mar-2021 04:54:55 PM CDT	Medium

Figure 18. 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.

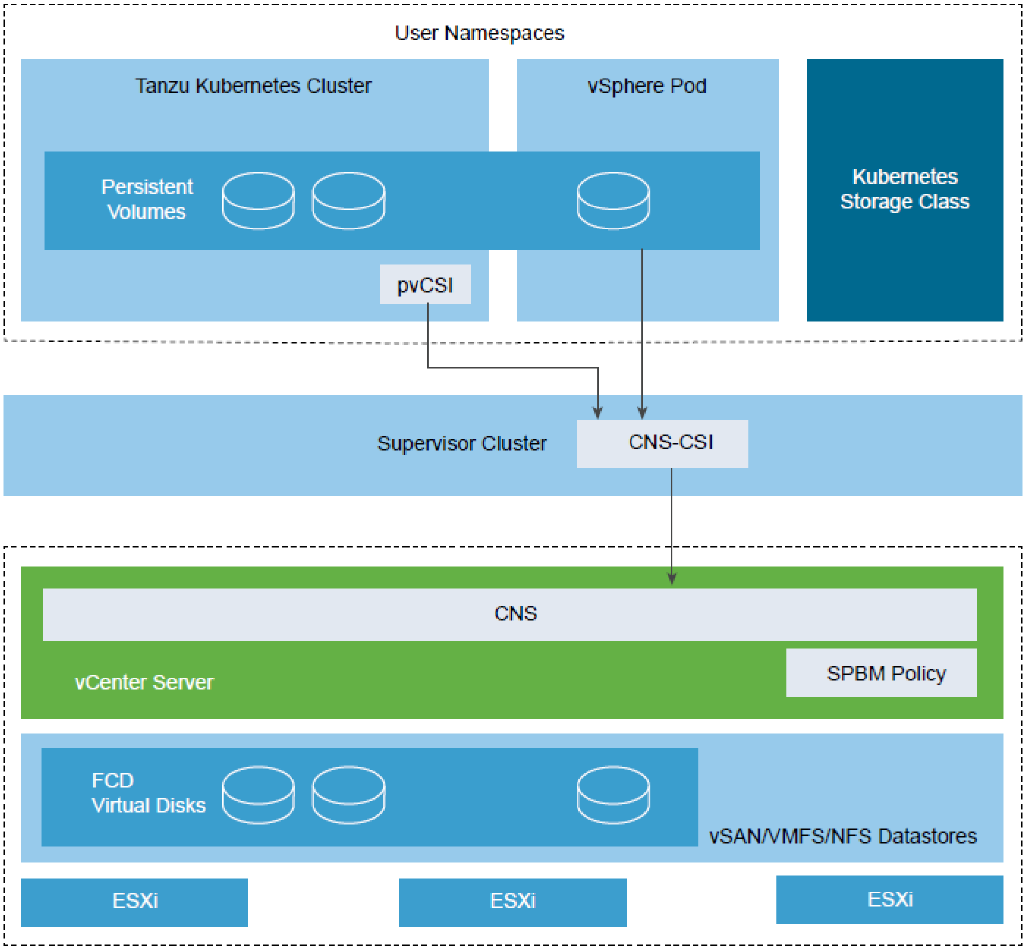


Figure 19. vSphere CNS-CSI and pvCSI architecture (credit: VMware)

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
```

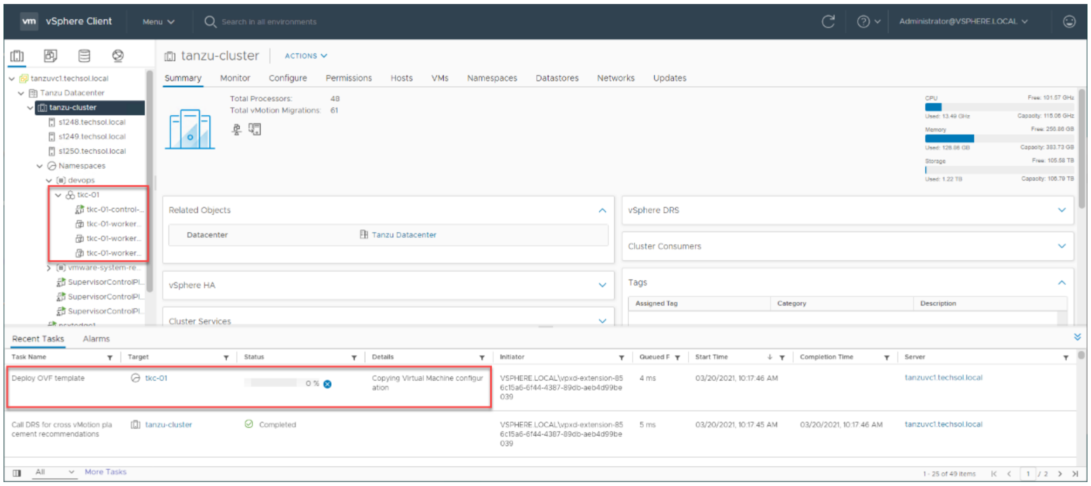
2. Use Kubernetes CLI Tools to log in and switch to the **devops** vSphere Namespace.

```
C:\>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".
```

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

```
C:\>kubectl 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.



Total Processors: 48
Total vMotion Migrations: 61

CPU Free: 101.57 GHz
Used: 13.49 GHz Capacity: 115.06 GHz
Memory Free: 256.86 GB
Used: 128.86 GB Capacity: 383.73 GB
Storage Free: 105.58 TB
Used: 1.22 TB Capacity: 106.79 TB

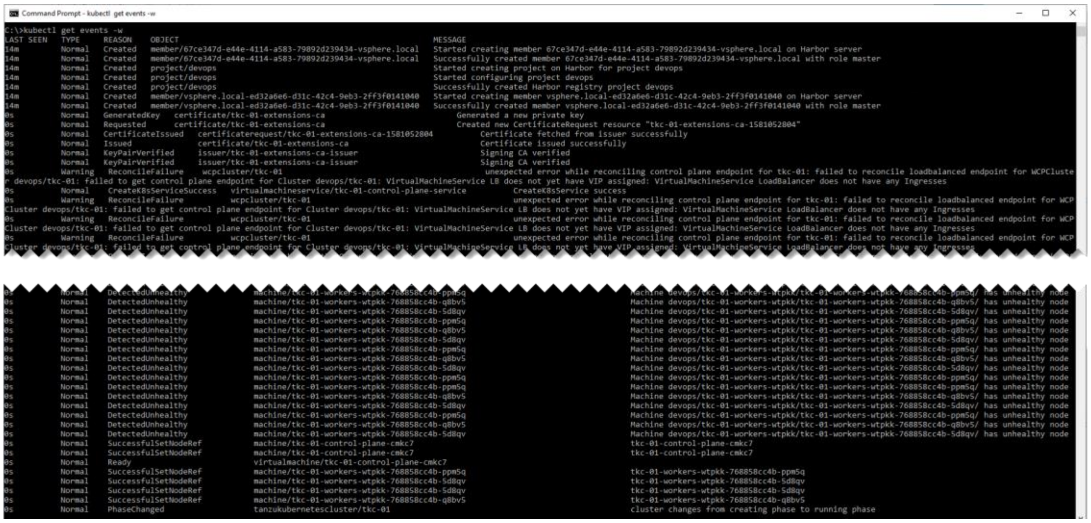
Datacenter	Tanzu Datacenter

Assigned Tag	Category	Description

Task Name	Target	Status	Details	Initiator	Queued F	Start Time	Completion Time	Server
Deploy OVF template	tkc-01	0 %	Copying Virtual Machine configuration	VSPHERE.LOCAL\vpxd-extension-856c15a6-6f44-4387-89db-aeb4d99be039	4 ms	03/20/2021, 10:17:46 AM		tanzuvc1.techsol.local
Call DRS for cross vMotion placement recommendations	tanzu-cluster	Completed		VSPHERE.LOCAL\vpxd-extension-856c15a6-6f44-4387-89db-aeb4d99be039	5 ms	03/20/2021, 10:17:45 AM	03/20/2021, 10:17:46 AM	tanzuvc1.techsol.local

Figure 20. 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.

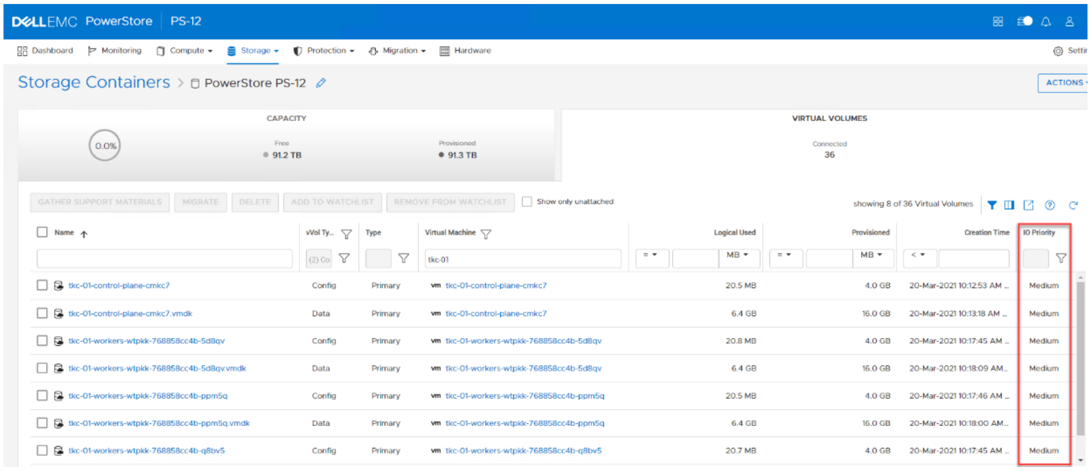


```
C:\>kubectl get events -w
LAST SEEN   TYPE     REASON    OBJECT                                                   MESSAGE
14m         Normal   Created   member/67ce347d-e44e-4114-a583-79892d239434-vsphere.local   Started creating member 67ce347d-e44e-4114-a583-79892d239434-vsphere.local on Harbor server
14m         Normal   Created   member/67ce347d-e44e-4114-a583-79892d239434-vsphere.local   Successfully created member 67ce347d-e44e-4114-a583-79892d239434-vsphere.local with role master
14m         Normal   Created   project/devops                                           Started creating project on Harbor for project devops
14m         Normal   Created   project/devops                                           Started configuring project devops
14m         Normal   Created   project/devops                                           Successfully created Harbor registry project devops
14m         Normal   Created   member/vsphere.local-ed32a6e6-d31c-42c4-9eb3-2ff3f0141040   Started creating member vsphere.local-ed32a6e6-d31c-42c4-9eb3-2ff3f0141040 on Harbor server
14m         Normal   Created   member/vsphere.local-ed32a6e6-d31c-42c4-9eb3-2ff3f0141040   Successfully created member vsphere.local-ed32a6e6-d31c-42c4-9eb3-2ff3f0141040 with role master
0s          Normal   GeneratedKey   certificate/tkc-01-extensions-ca                         Generated a new private key
0s          Normal   Requested      certificate/tkc-01-extensions-ca                         Created new CertificateRequest resource "tkc-01-extensions-ca-1581052804"
0s          Normal   CertificateIssued   certificaterequest/tkc-01-extensions-ca-1581052804         Certificate fetched from issuer successfully
0s          Normal   Issued              certificate/tkc-01-extensions-ca                           Certificate issued successfully
0s          Normal   KeyPairVerified     issuer/tkc-01-extensions-ca-issuer                         Signing CA verified
0s          Normal   KeyPairVerified     issuer/tkc-01-extensions-ca-issuer                         Signing CA verified
0s          Warning   ReconcileFailure    wcpcluster/tkc-01                                         unexpected error while reconciling control plane endpoint for tkc-01: failed to reconcile loadbalanced endpoint for WCPCluste
r devops/tkc-01: failed to get control plane endpoint for Cluster devops/tkc-01: VirtualMachineService LB does not yet have VIP assigned: VirtualMachineService LoadBalancer does not have any Ingresses
0s          Normal    CreateK8sServiceSuccess   virtualmachineservice/tkc-01-control-plane-service          CreateK8sService success
0s          Warning   ReconcileFailure          wcpcluster/tkc-01                                           unexpected error while reconciling control plane endpoint for tkc-01: failed to reconcile loadbalanced endpoint for WCP
Cluster devops/tkc-01: failed to get control plane endpoint for Cluster devops/tkc-01: VirtualMachineService LB does not yet have VIP assigned: VirtualMachineService LoadBalancer does not have any Ingresses
0s          Warning   ReconcileFailure          wcpcluster/tkc-01                                           unexpected error while reconciling control plane endpoint for tkc-01: failed to reconcile loadbalanced endpoint for WCP
Cluster devops/tkc-01: failed to get control plane endpoint for Cluster devops/tkc-01: VirtualMachineService LB does not yet have VIP assigned: VirtualMachineService LoadBalancer does not have any Ingresses
0s          Warning   ReconcileFailure          wcpcluster/tkc-01                                           unexpected error while reconciling control plane endpoint for tkc-01: failed to reconcile loadbalanced endpoint for WCP
Cluster devops/tkc-01: failed to get control plane endpoint for Cluster devops/tkc-01: VirtualMachineService LB does not yet have VIP assigned: VirtualMachineService LoadBalancer does not have any Ingresses
```

```
0s          Normal    DetectedUnhealthy         machine/tkc-01-workers-wtpkk-768858cc4b-ppm5q               Machine devops/tkc-01-workers-wtpkk/tkc-01-workers-wtpkk-768858cc4b-ppm5q/ has unhealthy node
0s          Normal    DetectedUnhealthy         machine/tkc-01-workers-wtpkk-768858cc4b-q8bv5               Machine devops/tkc-01-workers-wtpkk/tkc-01-workers-wtpkk-768858cc4b-q8bv5/ has unhealthy node
0s          Normal    DetectedUnhealthy         machine/tkc-01-workers-wtpkk-768858cc4b-5d8qv               Machine devops/tkc-01-workers-wtpkk/tkc-01-workers-wtpkk-768858cc4b-5d8qv/ has unhealthy node
0s          Normal    DetectedUnhealthy         machine/tkc-01-workers-wtpkk-768858cc4b-ppm5q               Machine devops/tkc-01-workers-wtpkk/tkc-01-workers-wtpkk-768858cc4b-ppm5q/ has unhealthy node
0s          Normal    DetectedUnhealthy         machine/tkc-01-workers-wtpkk-768858cc4b-q8bv5               Machine devops/tkc-01-workers-wtpkk/tkc-01-workers-wtpkk-768858cc4b-q8bv5/ has unhealthy node
0s          Normal    DetectedUnhealthy         machine/tkc-01-workers-wtpkk-768858cc4b-5d8qv               Machine devops/tkc-01-workers-wtpkk/tkc-01-workers-wtpkk-768858cc4b-5d8qv/ has unhealthy node
0s          Normal    DetectedUnhealthy         machine/tkc-01-workers-wtpkk-768858cc4b-ppm5q               Machine devops/tkc-01-workers-wtpkk/tkc-01-workers-wtpkk-768858cc4b-ppm5q/ has unhealthy node
0s          Normal    DetectedUnhealthy         machine/tkc-01-workers-wtpkk-768858cc4b-q8bv5               Machine devops/tkc-01-workers-wtpkk/tkc-01-workers-wtpkk-768858cc4b-q8bv5/ has unhealthy node
0s          Normal    DetectedUnhealthy         machine/tkc-01-workers-wtpkk-768858cc4b-5d8qv               Machine devops/tkc-01-workers-wtpkk/tkc-01-workers-wtpkk-768858cc4b-5d8qv/ has unhealthy node
0s          Normal    DetectedUnhealthy         machine/tkc-01-workers-wtpkk-768858cc4b-ppm5q               Machine devops/tkc-01-workers-wtpkk/tkc-01-workers-wtpkk-768858cc4b-ppm5q/ has unhealthy node
0s          Normal    DetectedUnhealthy         machine/tkc-01-workers-wtpkk-768858cc4b-ppm5q               Machine devops/tkc-01-workers-wtpkk/tkc-01-workers-wtpkk-768858cc4b-ppm5q/ has unhealthy node
0s          Normal    DetectedUnhealthy         machine/tkc-01-workers-wtpkk-768858cc4b-q8bv5               Machine devops/tkc-01-workers-wtpkk/tkc-01-workers-wtpkk-768858cc4b-q8bv5/ has unhealthy node
0s          Normal    DetectedUnhealthy         machine/tkc-01-workers-wtpkk-768858cc4b-5d8qv               Machine devops/tkc-01-workers-wtpkk/tkc-01-workers-wtpkk-768858cc4b-5d8qv/ has unhealthy node
0s          Normal    DetectedUnhealthy         machine/tkc-01-workers-wtpkk-768858cc4b-ppm5q               Machine devops/tkc-01-workers-wtpkk/tkc-01-workers-wtpkk-768858cc4b-ppm5q/ has unhealthy node
0s          Normal    DetectedUnhealthy         machine/tkc-01-workers-wtpkk-768858cc4b-q8bv5               Machine devops/tkc-01-workers-wtpkk/tkc-01-workers-wtpkk-768858cc4b-q8bv5/ has unhealthy node
0s          Normal    DetectedUnhealthy         machine/tkc-01-workers-wtpkk-768858cc4b-5d8qv               Machine devops/tkc-01-workers-wtpkk/tkc-01-workers-wtpkk-768858cc4b-5d8qv/ has unhealthy node
0s          Normal    SuccessfulSetNodeRef      machine/tkc-01-control-plane-cmkc7                          tkc-01-control-plane-cmkc7
0s          Normal    SuccessfulSetNodeRef      machine/tkc-01-control-plane-cmkc7                          tkc-01-control-plane-cmkc7
0s          Normal    Ready                     virtualmachine/tkc-01-control-plane-cmkc7
0s          Normal    SuccessfulSetNodeRef      machine/tkc-01-workers-wtpkk-768858cc4b-ppm5q               tkc-01-workers-wtpkk-768858cc4b-ppm5q
0s          Normal    SuccessfulSetNodeRef      machine/tkc-01-workers-wtpkk-768858cc4b-5d8qv               tkc-01-workers-wtpkk-768858cc4b-5d8qv
0s          Normal    SuccessfulSetNodeRef      machine/tkc-01-workers-wtpkk-768858cc4b-5d8qv               tkc-01-workers-wtpkk-768858cc4b-5d8qv
0s          Normal    SuccessfulSetNodeRef      machine/tkc-01-workers-wtpkk-768858cc4b-q8bv5               tkc-01-workers-wtpkk-768858cc4b-q8bv5
0s          Normal    PhaseChanged              tanzukubernetescluster/tkc-01                               cluster changes from creating phase to running phase
```

Figure 21. 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.



Storage Containers > PowerStore PS-12

CAPACITY

0.0%

Free
91.2 TB

Provisioned
91.3 TB

VIRTUAL VOLUMES

Connected
36

□ Show only unattached

showing 8 of 36 Virtual Volumes

Name	vVol Ty...	Type	Virtual Machine	Logical Used	Provisioned	Creation Time	IO Priority
	(2) Co		tkc-01	MB	MB		
tkc-01-control-plane-cmkc7	Config	Primary	vm tkc-01-control-plane-cmkc7	20.5 MB	4.0 GB	20-Mar-2021 10:12:53 AM ...	Medium
tkc-01-control-plane-cmkc7.vmdk	Data	Primary	vm tkc-01-control-plane-cmkc7	6.4 GB	16.0 GB	20-Mar-2021 10:13:18 AM ...	Medium
tkc-01-workers-wtpkk-768858cc4b-5d8qv	Config	Primary	vm tkc-01-workers-wtpkk-768858cc4b-5d8qv	20.8 MB	4.0 GB	20-Mar-2021 10:17:45 AM ...	Medium
tkc-01-workers-wtpkk-768858cc4b-5d8qv.vmdk	Data	Primary	vm tkc-01-workers-wtpkk-768858cc4b-5d8qv	6.4 GB	16.0 GB	20-Mar-2021 10:18:09 AM...	Medium
tkc-01-workers-wtpkk-768858cc4b-ppm5q	Config	Primary	vm tkc-01-workers-wtpkk-768858cc4b-ppm5q	20.5 MB	4.0 GB	20-Mar-2021 10:17:46 AM ...	Medium
tkc-01-workers-wtpkk-768858cc4b-ppm5q.vmdk	Data	Primary	vm tkc-01-workers-wtpkk-768858cc4b-ppm5q	6.4 GB	16.0 GB	20-Mar-2021 10:18:00 AM...	Medium
tkc-01-workers-wtpkk-768858cc4b-q8bv5	Config	Primary	vm tkc-01-workers-wtpkk-768858cc4b-q8bv5	20.7 MB	4.0 GB	20-Mar-2021 10:17:45 AM ...	Medium

Figure 22. 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
VERSION
tkc-01-control-plane-cmkc7               Ready    master   76m
v1.16.14+vmware.1
tkc-01-workers-wtpkk-768858cc4b-5d8qv    Ready    <none>   72m
v1.16.14+vmware.1
tkc-01-workers-wtpkk-768858cc4b-ppm5q    Ready    <none>   72m
v1.16.14+vmware.1
```

```
tkc-01-workers-wtpkk-768858cc4b-q8bv5   Ready    <none>   72m
v1.16.14+vmware.1

C:\>kubectl config get-contexts
CURRENT   NAME                CLUSTER           AUTHINFO
NAMESPACE
          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.

VMware Tanzu Kubernetes Grid

Up to this point, this paper has covered vSphere with Tanzu (also known as TKGs) deployed on PowerStore T and PowerStore X models. 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 multi-cloud 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 a PowerStore T model for back end storage. TKG can also be deployed to the AppsOn PowerStore X model to fulfill end-to-end vSphere, compute, and storage needs.

When comparing vSphere with Tanzu to TKG, one key difference is the number of ESXi hosts required to deploy. vSphere with Tanzu requires a minimum of three ESXi hosts, meaning a minimum of two PowerStore X model appliances in a cluster. TKG can be deployed on two ESXi hosts, which means that a single PowerStore X model appliance consisting of two nodes will meet deployment requirements. With this in mind, TKG on a

PowerStore X model could be a good fit as an edge solution. For more information, see VMware Tanzu Edge Solution Architecture with ROBO Topology.

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 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 EMC PowerStore Virtualization Infrastructure Guide
- 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