WHITE PAPER – **MAY 2020**

VMWARE VSPHERE WITH KUBERNETES SUPPORT ON THE VMWARE CLOUD FOUNDATION MANAGEMENT DOMAIN

vmware®

Table of Contents

Announcing VMware vSphere with Kubernetes Support on the VMware Cloud Foundation Management Domain

When VMware Cloud Foundation™ 4.0 was released in April 2020, it introduced support for VMware vSphere® with Kubernetes. With the initial release, users were required to create a separate virtual infrastructure (VI) domain to host their Kubernetes workloads to enable vSphere with Kubernetes. This meant that at least seven servers were required to run vSphere with Kubernetes on Cloud Foundation: four hosts for the Cloud Foundation management domain and three additional hosts for a separate VI domain. VMware has now certified enablement of vSphere with Kubernetes on the management domain. This effectively reduces the minimum host count from seven to four.

With this change, you can deploy the Cloud Foundation consolidated architecture and enable vSphere with Kubernetes. You can get started with just four hosts; as your environment grows, you can easily scale up to the Cloud Foundation workload domain configuration maximums.

This paper provides an overview of how to enable vSphere with Kubernetes on the Cloud Foundation management domain.

vSphere with Kubernetes

vSphere with Kubernetes transforms vSphere clusters into a platform on which you can run Kubernetes workloads directly on VMware ESXi™ hosts and can create upstream Kubernetes clusters within dedicated resource pools, referred to as *namespaces*. This new capability builds on existing virtual machine (VM) capabilities of vSphere, providing a common "developer-ready" platform on which container-based workloads run alongside VM-based workloads on common infrastructure and with a common user interface (UI) and set of management tools.

Cloud Foundation

vSphere with Kubernetes is available as part of Cloud Foundation 4.0. Cloud Foundation is a hybrid cloud platform designed for running both traditional enterprise applications and modern applications. It is built on the proven and comprehensive software-defined VMware® stack, including vSphere with Kubernetes, VMware vSAN™, VMware NSX-T Data Center™, and VMware vRealize® Suite. Cloud Foundation provides a complete set of software-defined services for compute, storage, network security, Kubernetes management, and cloud management. The result is agile, reliable, efficient cloud infrastructure that offers consistent operations across private and public clouds.

Enabling vSphere with Kubernetes on the Cloud Foundation Management Domain

Enabling vSphere with Kubernetes on Cloud Foundation involves using the advanced automation available with Cloud Foundation to quickly stand up the VI, configure the VMware NSX® prerequisites, and enable vSphere with Kubernetes.

In this paper, the VI backing the vSphere with Kubernetes cluster is provided by the Cloud Foundation management domain. The management domain is created by the VMware Cloud Builder appliance during an initial deployment process referred to as *bring-up*. It is assumed that Cloud Foundation has been deployed with a single management domain that comprises one vSphere cluster. The procedure discussed here begins after bring-up and covers the following activities to be performed by the vSphere cloud administrator:

1. Deploy a VMware NSX Edge™ cluster in the management domain
2. Configure the NSX Edge cluster for vSphere with Kubernetes
3. Enable vSphere with Kubernetes on the management domain
4. Create a content library on the management domain VMware vCenter Server® instance
5. Enable the Harbor image registry on the management domain cluster
6. Create a namespace and configure access to vSphere with Kubernetes

The following are requirements:

- Cloud Foundation 4.0 must be deployed (that is, bring-up complete) with one vSphere cluster. The management domain should be in a healthy state.
- Application virtual networks (AVNs) cannot be deployed on the management domain. During bring-up, disable the AVN deployment on the deployment parameters spreadsheet.
- The vSphere cluster backing the management domain must have ample capacity for hosting both the Cloud Foundation infrastructure workloads (vCenter Server instances, VMware NSX Manager™ instances, SDDC Manager, and so on) and the vSphere with Kubernetes workloads (Kubernetes supervisor cluster, Harbor image registry, deployed Pods, and Tanzu Kubernetes Grid (TKG) clusters). If additional capacity is required, use the SDDC Manager to add hosts to the management domain cluster prior to enabling vSphere with Kubernetes.
- vSAN is required on the Cloud Foundation management domain. At least four VMware vSAN ReadyNodes™ are required.

Prerequisites for Enabling vSphere with Kubernetes

Prior to enabling vSphere with Kubernetes, deploy an NSX Edge cluster with the following specifications. These settings are a hard requirement for enabling vSphere with Kubernetes:

- Two edge transport nodes
- Large form factor
- Active/active configuration

Deploy the NSX Edge Cluster

To deploy the NSX Edge cluster, perform the following steps:

1. Add FQDN entries in DNS for the two edge transport nodes.
2. Deploy the NSX Edge cluster using the SDDC Manager.
3. Enable trust on the management cluster vCenter Server instance.
4. Verify that the "WCPReady" tag is set on the NSX Edge cluster.
5. Implement the custom route map workaround.

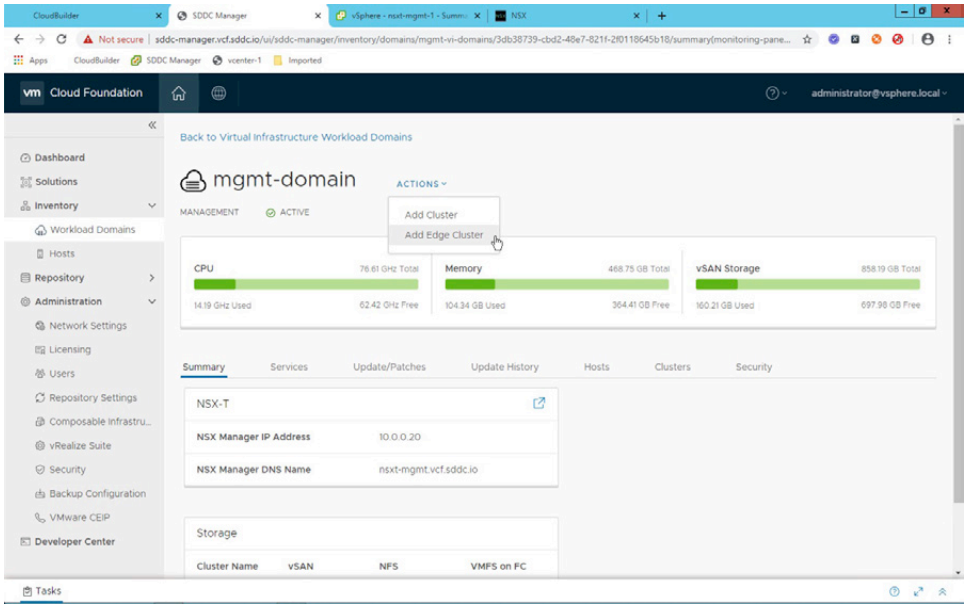
Step 1: Add FQDN for the edge transport nodes.

Begin by adding DNS records for the two edge transport nodes. Add both forward (A) and reverse (PTR) lookup entries. In the following examples, we are naming the edge transport nodes *edge01-mgmt.vcf.sddc.lab* and *edge02-mgmt.vcf.sddc.lab*. These nodes use the IPs *10.0.0.51* and *10.0.0.52* respectively for their management interface.

```
# DNS Forward Lookup Records

     edge01-mgmt.vcf.sddc.lab. IN A 10.0.0.51

     edge02-mgmt.vcf.sddc.lab. IN A 10.0.0.52

# DNS Reverse Lookup Records

     51.0 IN PTR edge01-mgmt.vcf.sddc.lab.

     52.0 IN PTR edge02-mgmt.vcf.sddc.lab.
```

Step 2: Deploy the NSX Edge cluster using the SDDC Manager.

Deploying an NSX Edge cluster is an automatic operation in Cloud Foundation. Log in to the SDDC Manager. From the dashboard, navigate to **Workload Domains** -> **mgmt-domain**; under **ACTIONS**, select **Add Edge Cluster**.

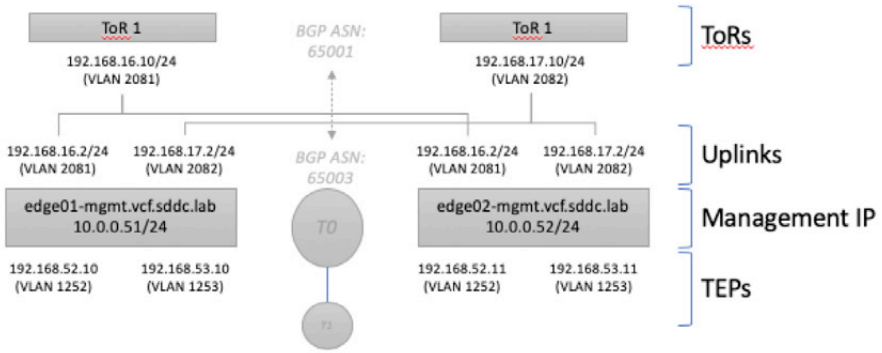


You are first presented with a summary of prerequisites that must be completed. Have the following information on hand to complete the workflow:

- FQDN for the edge transport nodes (ensure that DNS records have been added)
- IP addresses for the edge transport node management network
- VLAN ID and IP addresses for the edge transport node tunnel endpoint IPs (TEPs)
- VLAN ID and IP addresses for the two uplink networks
- (When using BGP) The BGP ASN and peering information

NOTE: In this paper, I use BGP. If you want to use static routing, see this blog from Cormac Hogan.

A walkthrough demo showing the steps to deploy an NSX Edge cluster on Cloud Foundation is available at the VMware Cloud Foundation Resource Center. It is recommended that you create a diagram similar to the following one to help you understand the networking requirements and to provide a reference to assist with troubleshooting.



Step 3: Enable trust on the vCenter Server instance.

Enable the NSX Edge Cluster for vSphere with Kubernetes

For the management cluster to be recognized as a "compatible cluster" in the vSphere Web Client, you must update the NSX configuration to enable trust on the vCenter Server instance and add the "WCPReady" tag to the NSX Edge cluster.

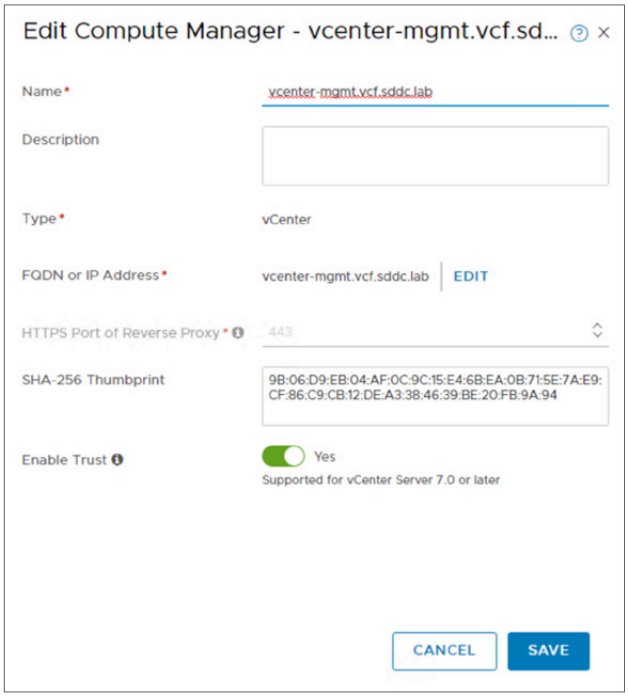
To enable "trust" on the vCenter Server instance, perform the following steps:

Log in to the **NSX Manager** instance.

Navigate to **System** -> **Fabric** -> **Compute Managers.**

Select the **vCenter Server** instance and click **EDIT.**

Toggle **Enable Trust** to **Yes.**



Step 4: Verify the "WCPReady" tag.

To identify vSphere clusters that are eligible for vSphere with Kubernetes, a "WCPReady" tag is assigned to the NSX Edge cluster. If you deploy the NSX Edge cluster from the SDDC Manager using the "Workload Management" option, this tag is created automatically. If you choose the "Custom" use-case option, or to manually deploy the NSX Edge cluster, you must set this tag manually. To verify the "WCPReady" tag on the NSX Edge cluster, perform the following steps:

Log in to the **NSX Manager** instance.

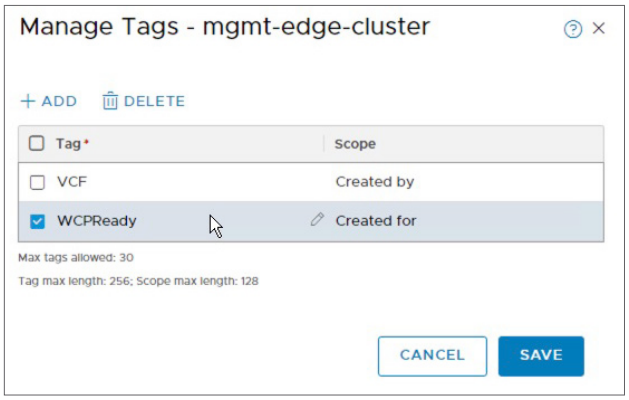
Navigate to **System** -> **Fabric** -> **Nodes** -> **Edge Clusters.**

Click the **Edge Cluster** name.

Next to **Tags**, click **MANAGE.**

Verify/Add the tag: **WCPReady/Created for.**

If the tag is missing, add it.



Step 5: Implement a custom route map.

There is a bug in Cloud Foundation 4.0/NSX-T Data Center 3.0 whereby the BGP route advertisements for networks attached to the Tier-1 logical router are blocked by default. For these routes to be advertised to the top-of-rack (TOR) switches, apply the following workaround on the Tier-0 logical router. For more information, refer to Configure NSX Route Maps on Edge T-0 Router in the Cloud Foundation documentation.

Add a new IP prefix with the name **Any network** that will permit all networks:

Log in to the **NSX-T Manager** instance.

Navigate to **Networking -> Tier-0 Gateways.**

Click the vertical ellipses and select **EDIT.**

Expand the **ROUTING** section.

Select the **IP Prefix List** hyperlink.

Click **ADD IP PREFIX LIST.**

Enter **Any network** for the name.

Click **SET.**

Click **ADD PREFIX.**

Enter **Any** for the CIDR.

Toggle **ACTION** to **PERMIT.**

Click **ADD.**

Click **APPLY.**

Click **SAVE.**

Click **CLOSE.**

Select the **Route Maps** hyperlink.

Click **ADD ROUTE MAP.**

Enter the name **Custom Route Map.**

Click **SET.**

Click **ADD MATCH CRITERIA.**

Next to **IP Prefix,** click **SET.**

Select the **Any network** IP prefix.

Click **SAVE.**

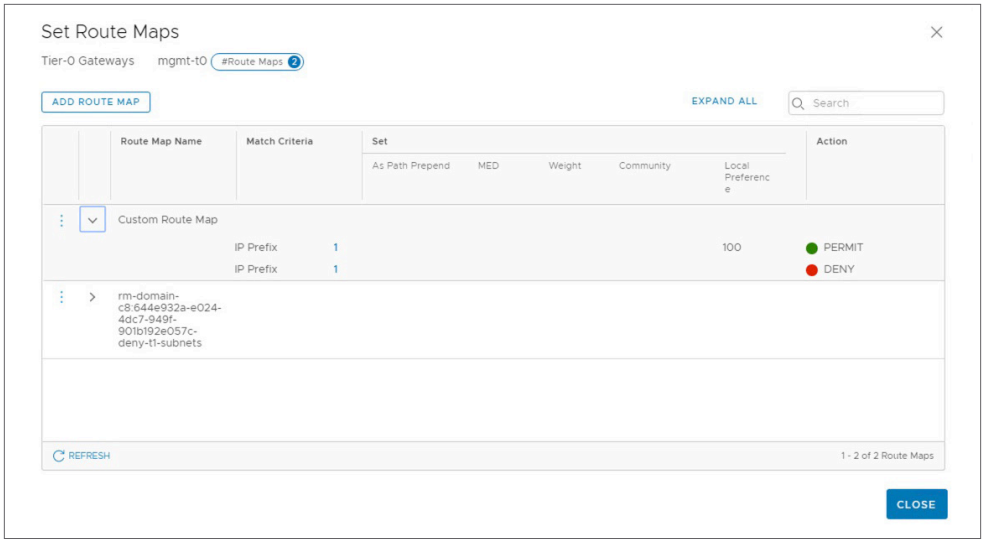
Set **ACTION** to **PERMIT.**

Click **ADD.**

Click **APPLY.**

Click **SAVE.**

Click **CLOSE.**



Set the Tier-0 **Route Re-distribution** to use the custom route map.

Expand **Route Re-distribution.**

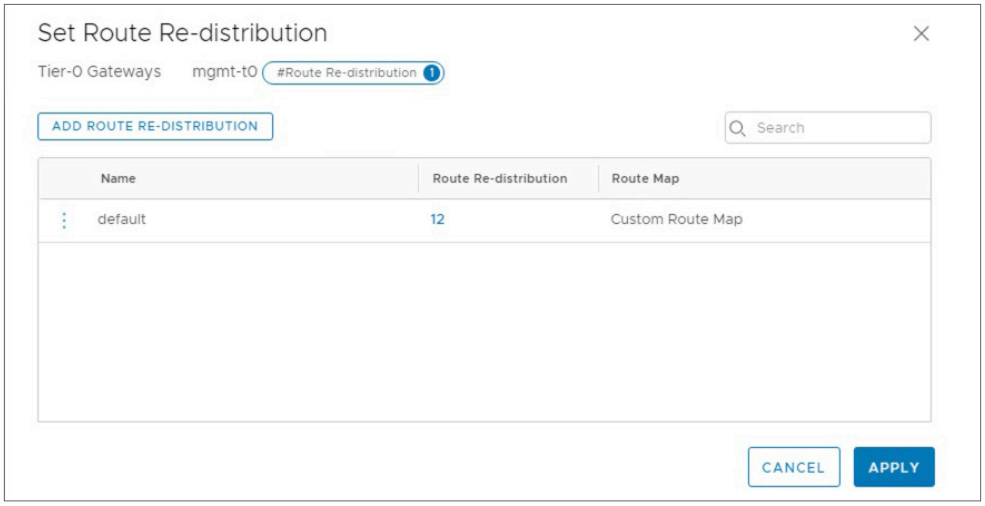
Click the hyperlink next to **Route Re-distribution.**

Click the vertical ellipses and select **EDIT.**

Set the **Route Map** to **Custom Route Map.**

Click **ADD ROUTE RE-DISTRIBUTION.**

Click **APPLY.**



With the new route map, the Kubernetes networks connected to the Tier-1 logical router will now be advertised to the upstream routers. Filters can still be applied in the BGP neighbor configuration if necessary.

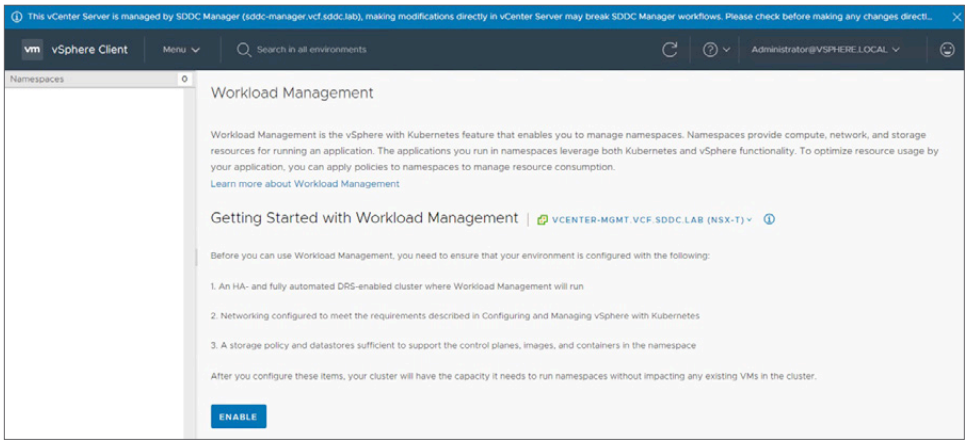
Enable vSphere with Kubernetes

After the NSX Edge cluster has been deployed, you are ready to enable vSphere with Kubernetes. Normally, you would enable vSphere with Kubernetes from the SDDC Manager. However, in Cloud Foundation 4.0, the SDDC Manager excludes the management domain from the list of available clusters in the UI. To enable vSphere with Kubernetes on the management domain, you must use the vSphere Web Client instance.

Enabling vSphere with Kubernetes involves selecting the cluster where you want to enable Kubernetes and providing details related to the network and storage configuration of your environment. An overview is provided in the section that follows. A detailed explanation of the input parameters required to enable vSphere with Kubernetes is out of scope for this paper. To learn more about the input parameters required to enable vSphere with Kubernetes, refer to the Cloud Foundation documentation.

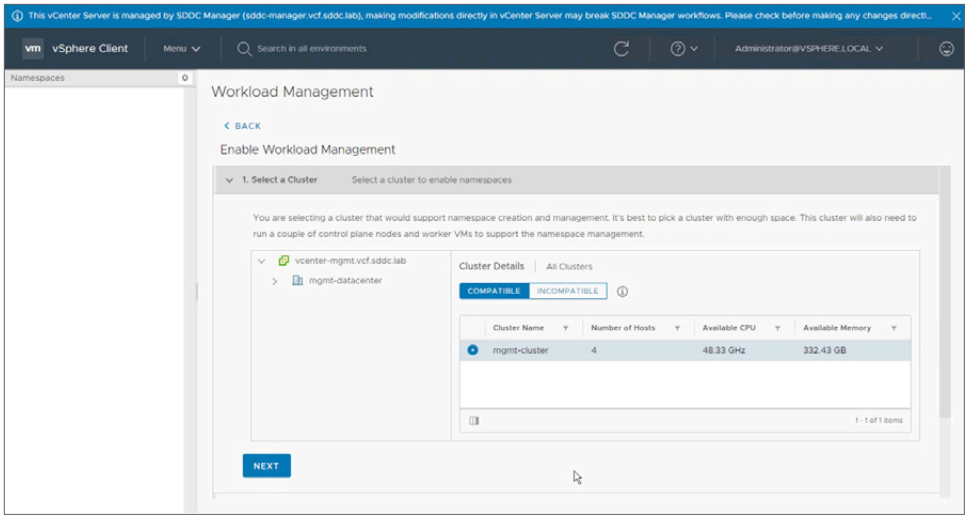
Log in to the **vSphere Web Client** instance.

Navigate to **Menu** –> **Workload Management.**



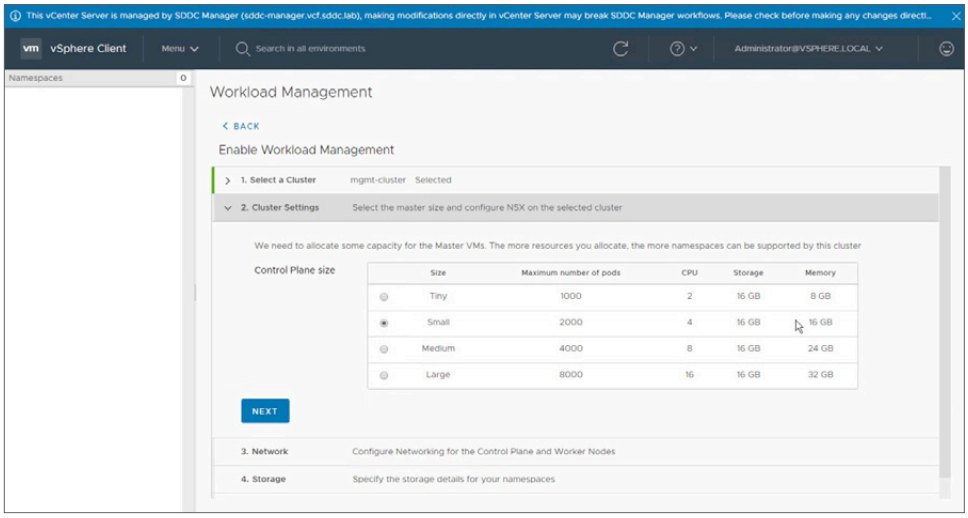
A list of requirements is displayed.

Click **ENABLE.**



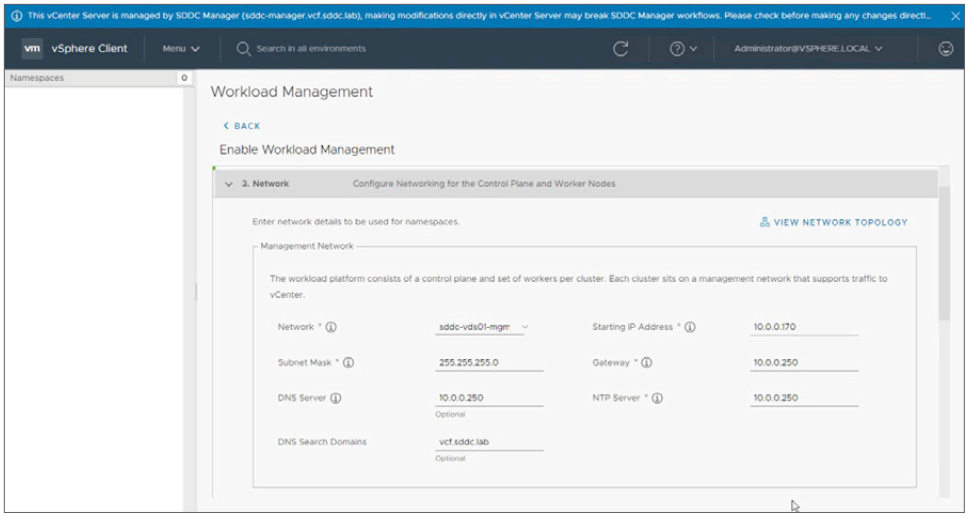
Select **mgmt-cluster.**

Click **NEXT.**



Set the **Control Plane size.**

Click **NEXT.**

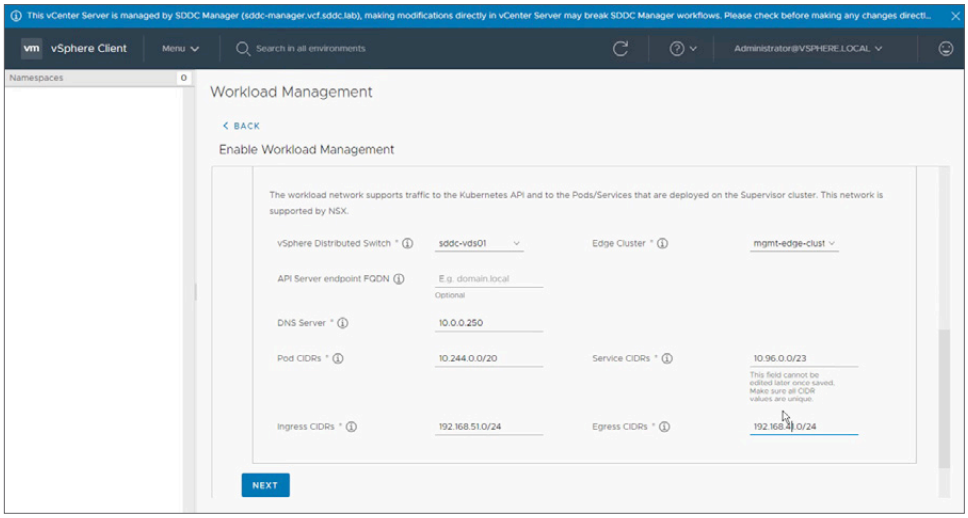


Enter the **Management Network** details. This includes the following steps:

Select the cluster management **Network.**

Enter the **Starting IP Address** in the range of five consecutive IPs for the Kubernetes supervisor cluster.

Enter the **Subnet Mask, Gateway, DNS Server,** and **NTP Server** addresses.



Enter the networking details for the **Kubernetes** control plane.

This includes the following steps:

Select the VMware **vSphere Distributed Switch™** to use on the cluster.

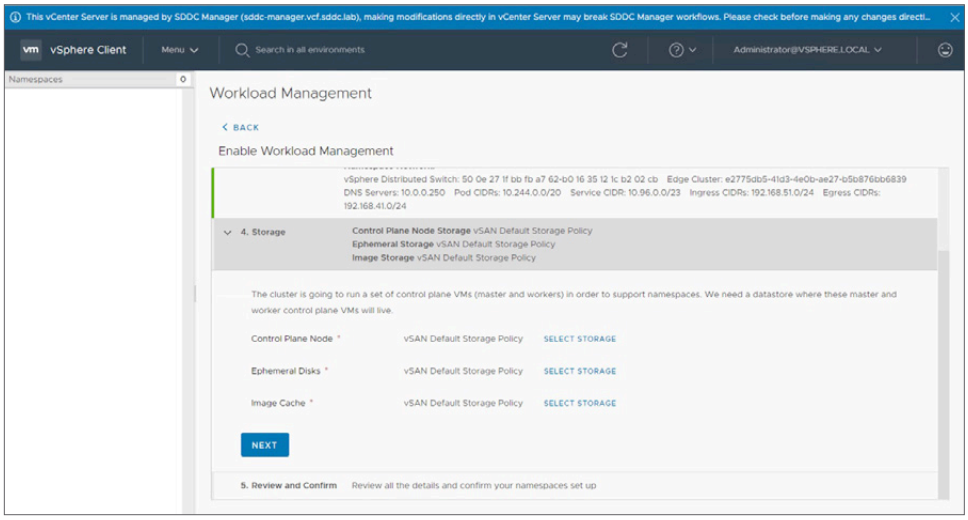
Select the **Edge Cluster.**

Provide the **DNS Server.**

Enter the **Pod CIDRs** and **Service CIDRs.**

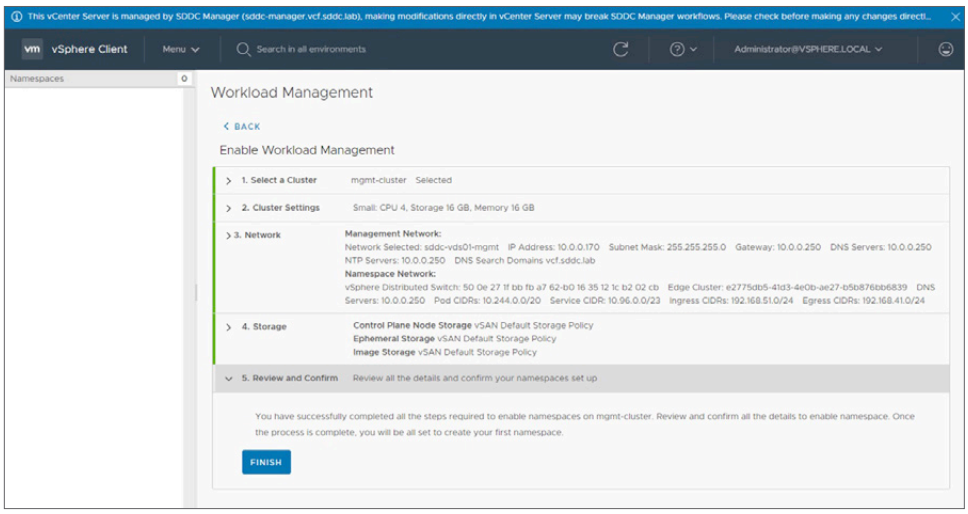
Enter the **Ingress CIDRs** and **Egress CIDRs.**

Click **NEXT.**

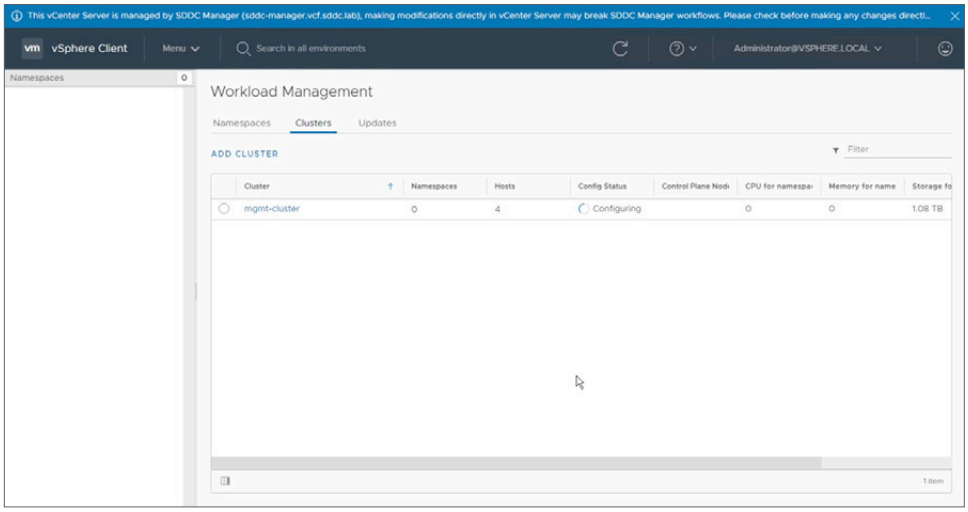


Select the **vSAN Storage Policy** to use for the **Control Plane Node, Ephemeral Disks,** and the **Image Cache.**

Click **NEXT.**



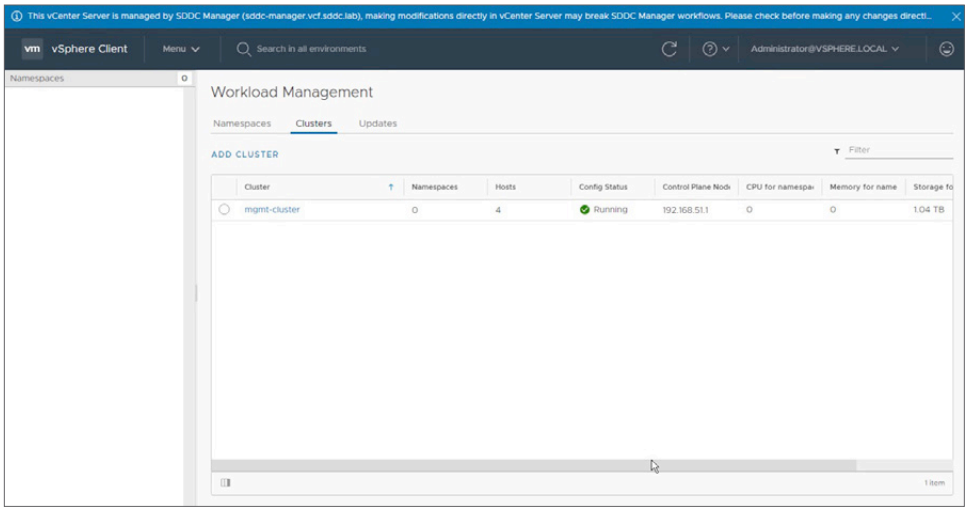
Review the input parameters; when ready to enable vSphere with Kubernetes, click **FINISH.**



It takes approximately 10 minutes to create the Kubernetes supervisor cluster and enable vSphere with Kubernetes on the management domain. During this time, the following high-level tasks are performed on the cluster:

- The "container runtime" (CRE) and "spherelet" binaries are pushed out to the ESXi hosts.
- Three Kubernetes supervisor nodes are deployed, and the vSphere Pod service is instantiated.
- The Tier-0 and Tier-1 logical routers, and their related load balancer and NAT services, are configured for use with vSphere with Kubernetes.

The cluster status shows **Running** when vSphere with Kubernetes has been successfully enabled.



At this point, vSphere with Kubernetes has been enabled. However, there are additional steps that must be performed before you are ready to hand off the cluster to the developers. These include the following:

- Create a content library.
- Deploy the Harbor image registry.
- Create a namespace and configuring access.

Create a Content Library

vSphere with Kubernetes uses the vSphere content library to store VM templates used to deploy Tanzu Kubernetes Grid (TKG) clusters.

You can choose to manually upload the TKG VM templates, or you can subscribe to a VMware hosted repository to download the VM templates. In this example, we subscribe to the VMware hosted repository using the subscription URL https://wp-content.vmware.com/v2/latest/lib.json.

To add a content library, perform the following steps:

Log in to the **vSphere Web Client** instance.

Navigate **Home** -> **Content Libraries.**

Click **+Create.**

Name = Kubernetes.

vCenter Server = **vcenter-1.vcf.sddc.lab.**

Click **NEXT.**

Select **Subscribed Content Library.**

Subscription URL = https://wp-content.vmware.com/v2/latest/lib.json.

Click **NEXT.**

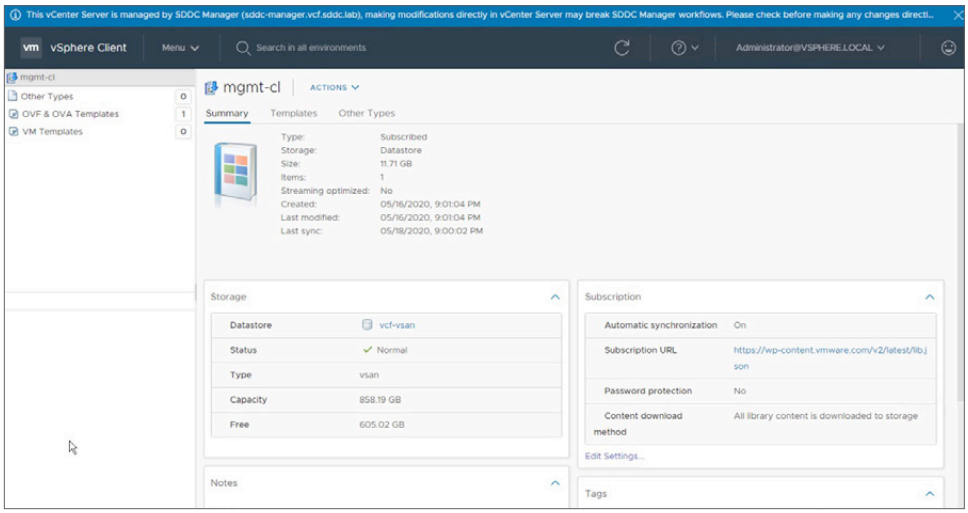
Click **Yes** when asked to verify authenticity.

Select **vcf-vsan.**

Click **NEXT.**

Click **FINISH.**

It takes only a minute to create the content library. However, it can take several minutes for the VM template images to download.



Deploy the Harbor Registry

Enable a private image registry on the supervisor cluster by using the built-in Harbor registry service. Developers can push and pull images from the registry, where they can be used to deploy vSphere Pods.

To enable the Harbor registry, perform the following steps:

From the **vSphere Web Client** instance, navigate to the **Host** and **Clusters** view.

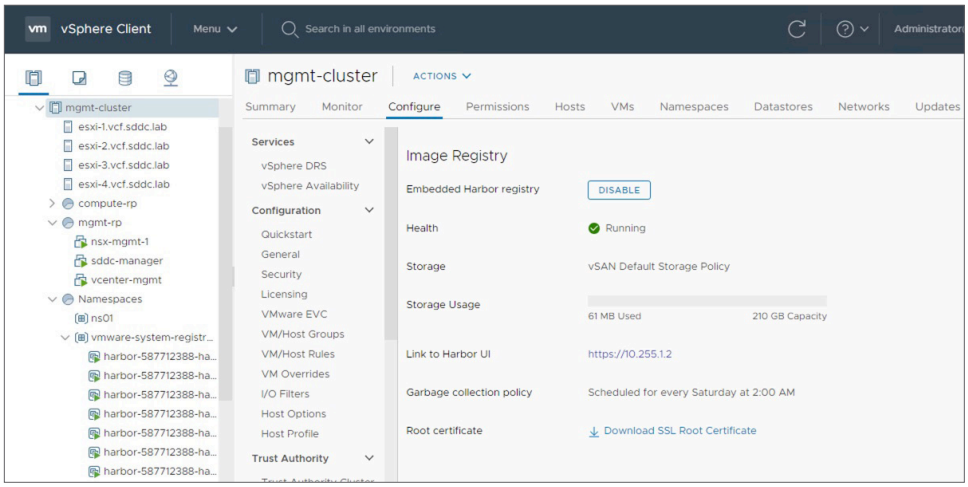
Select the **Cluster** in the management domain.

Select **Configure** tab.

Scroll down and select **Image Registry.**

Click **ENABLE.**

It takes approximately 15 minutes for the Harbor registry to deploy. When deployed, the health status shows **Running** and you will be presented with the **Link to Harbor UI**.



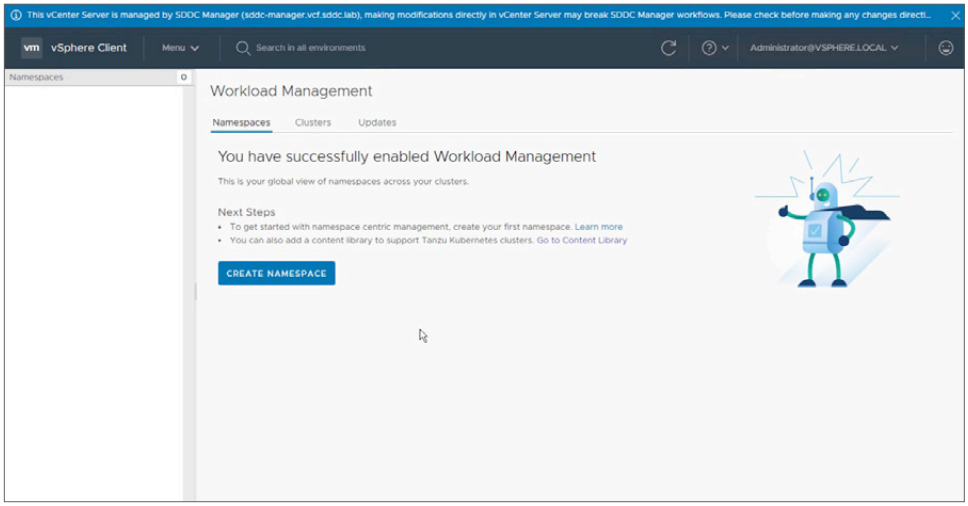
Create a vSphere Namespace

Namespaces are used to manage user access and control resource consumption within your Kubernetes enabled cluster. Use the vSphere Web Client instance to create namespaces on the supervisor cluster. When created, assign access and define resource limits.

To create a namespace, perform the following steps:

From the **vSphere Web Client** instance, select **Home** -> **Workload Management** -> **Namespaces.**

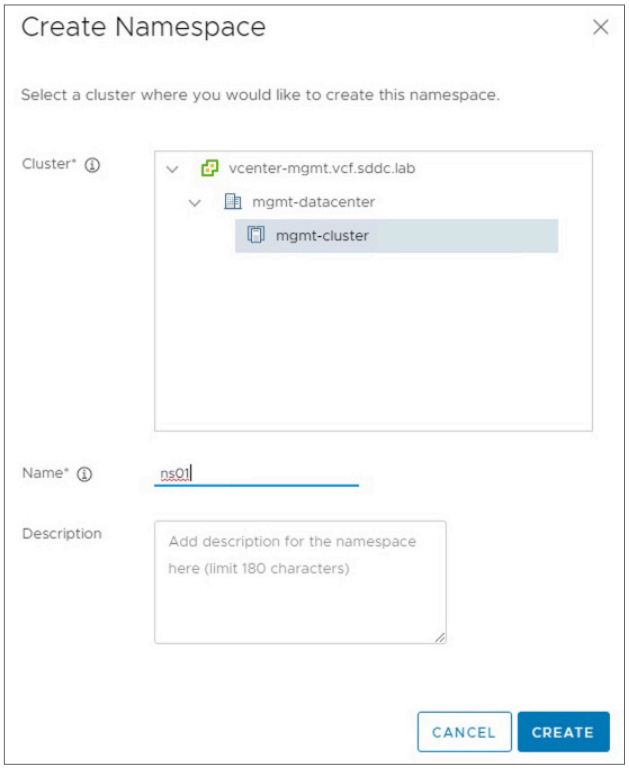
Click **CREATE NAMESPACE.**



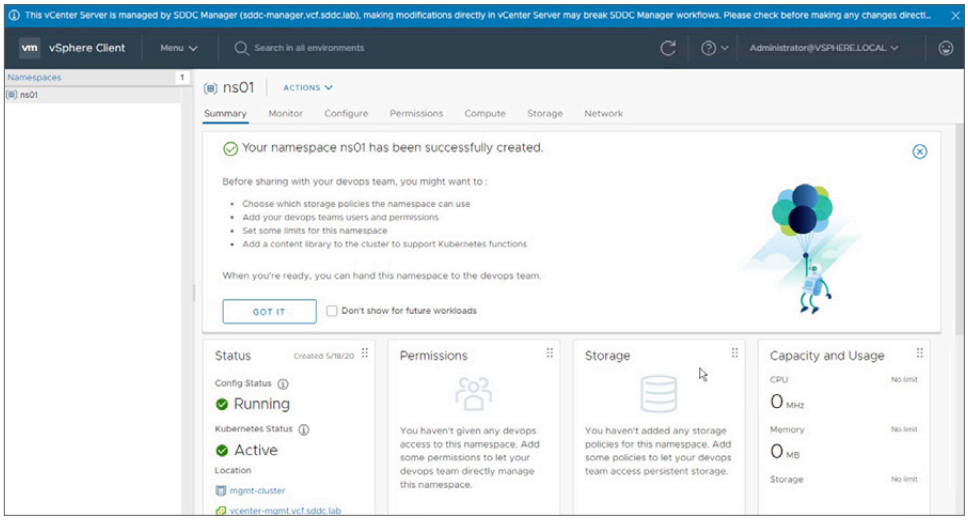
Expand the tree and select **mgmt-cluster.**

Enter a **Name.**

Click **CREATE.**



This is a quick operation that will complete in a few seconds. You will be notified that the namespace has been created and will be provided with a list of next steps.



Step 1: Enable access to the namespace.

vSphere with Kubernetes uses single sign on (SSO) to authenticate users and grant access to namespaces. Typically, customers add their Microsoft Active Directory (AD) domain as an identity source in SSO. Users authenticate using their AD credentials. In this example, I create a simple user account (ava@vsphere.local) and group (devteam) to the default SSO domain **vsphere.local**.

Step 2: Add user accounts.

From the **vSphere Web Client** instance, navigate **Home** ->

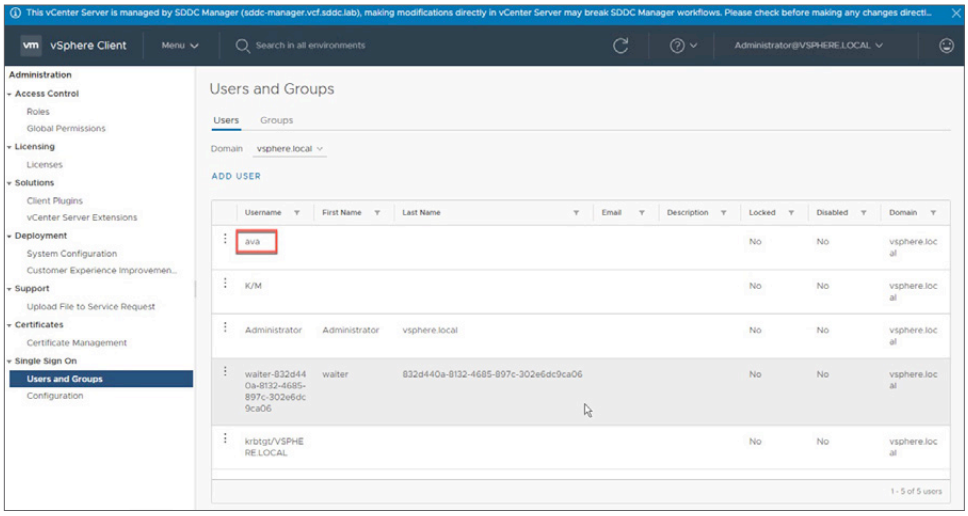
Administration -> **Single Sign On** -> **Users and Groups.**

From the **Users** tab, set **Domain** to **vsphere.local.**

Click **ADD USER.**

Enter the user ava@vsphere.local and set a password.

Click **ADD.**



Step 3: Add group.

From the **vSphere Web Client** instance, navigate **Home** ->

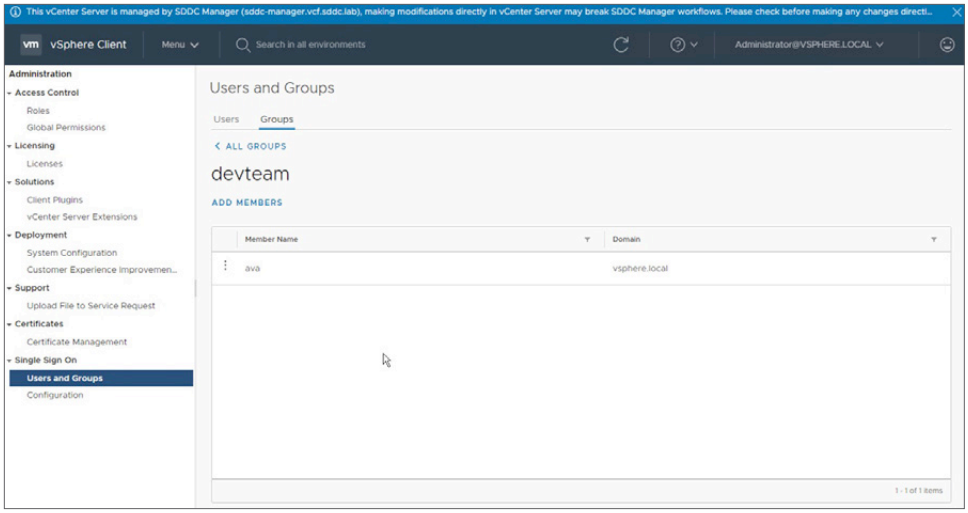
Administration -> **Single Sign On** -> **Users and Groups.**

From the **Groups** tab, click **ADD GROUP.**

Enter the **Group Name, devteam.**

Add the user **ava@vsphere.local.**

Click **ADD.**



Step 4: Configure a namespace.

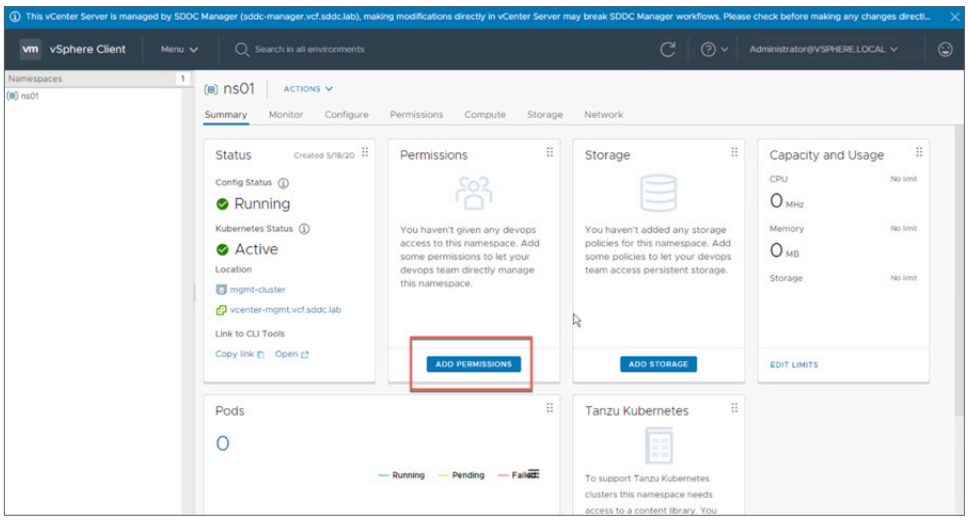
Prior to configuring a namespace, the following requirements must be met:

1. Content library created
2. Harbor registry enabled
3. User and groups defined in the vsphere.local SSO domain
4. Namespace created

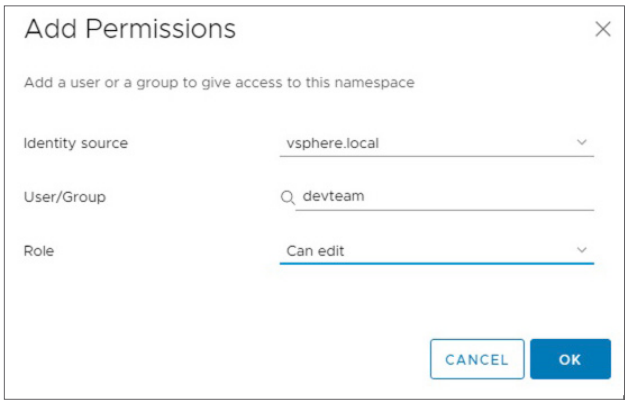
To configure a namespace, perform the following steps:

From the **vSphere Web Client** instance, navigate **Home –> Workload Management.**

Select the **Namespace.**



Click **ADD PERMISSIONS.**

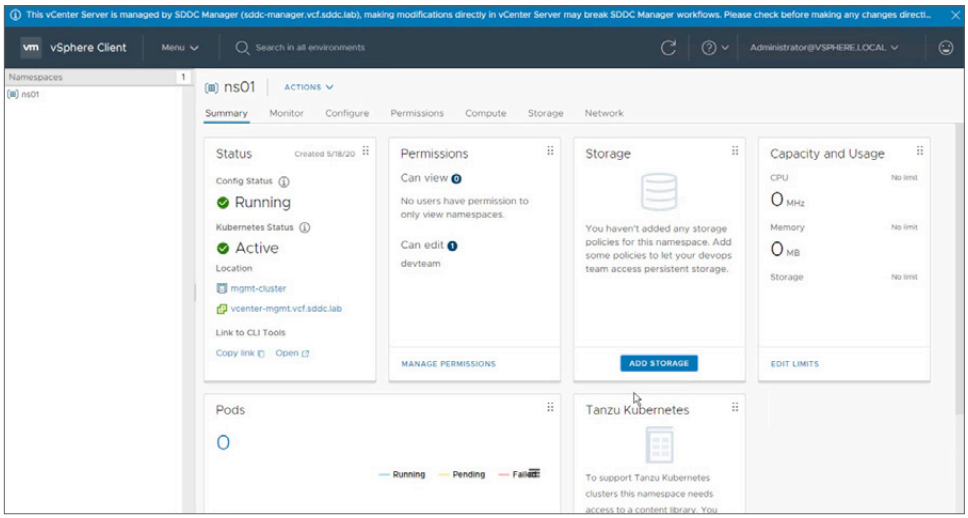


Set **Identity source** to **vsphere.local.**

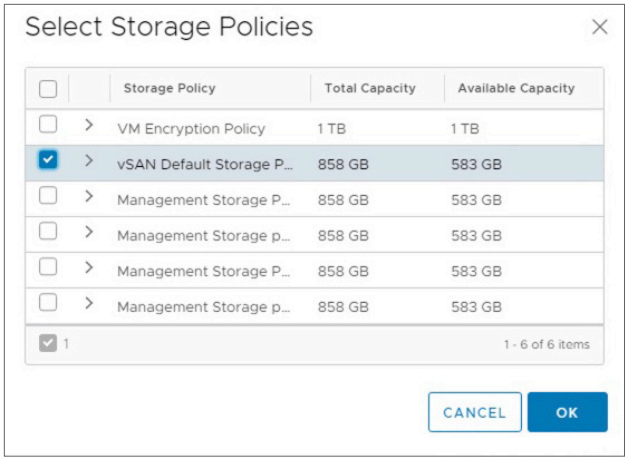
Set **User/Group** to **devteam.**

Set **Role** to **Can edit.**

Click **OK.**

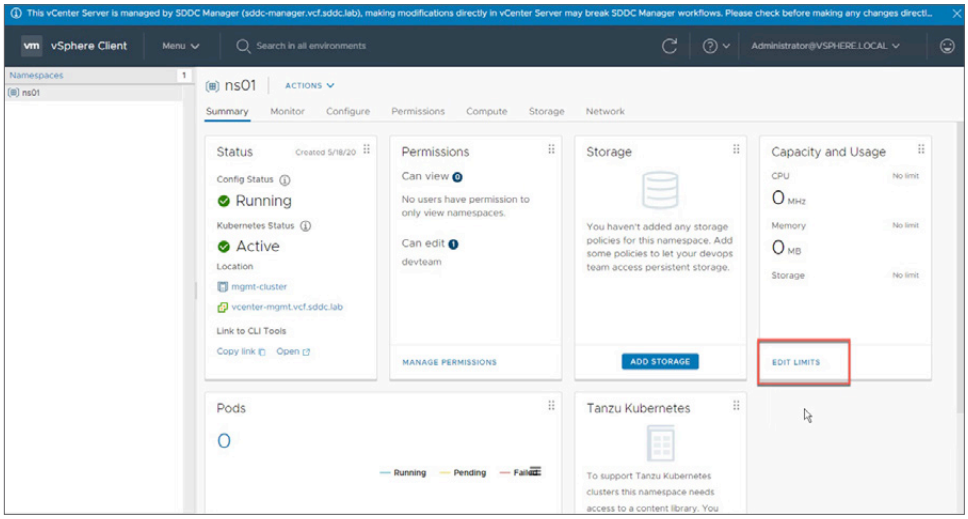


Click **ADD STORAGE.**

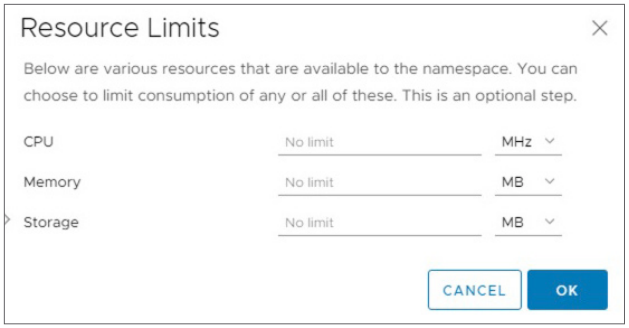


Select the preferred **Storage Policies.**

Click **OK.**

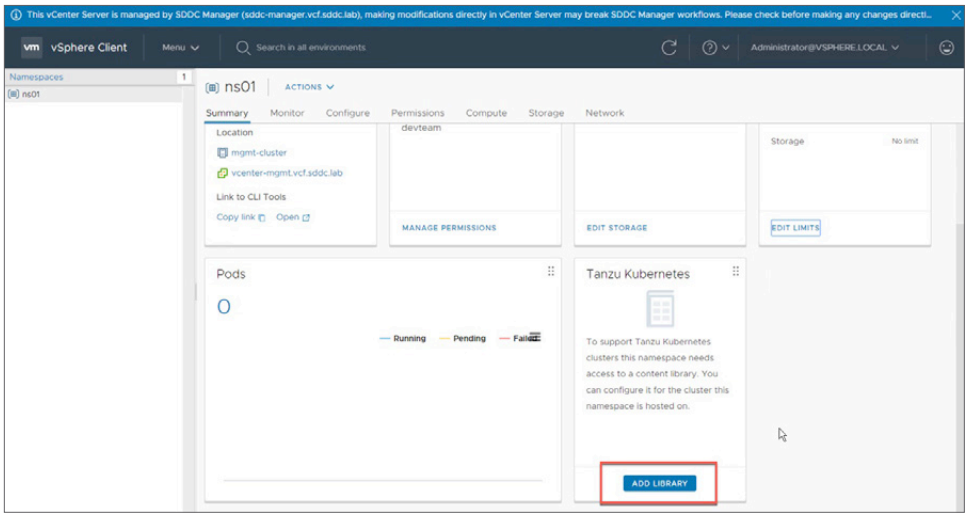


Click **EDIT LIMITS.**

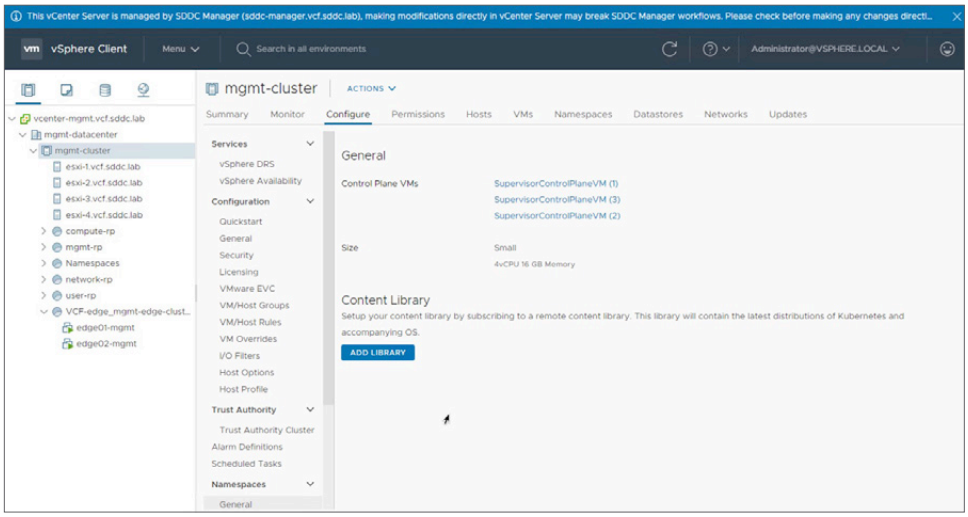


Set the **CPU**, **Memory**, and **Storage** limits.

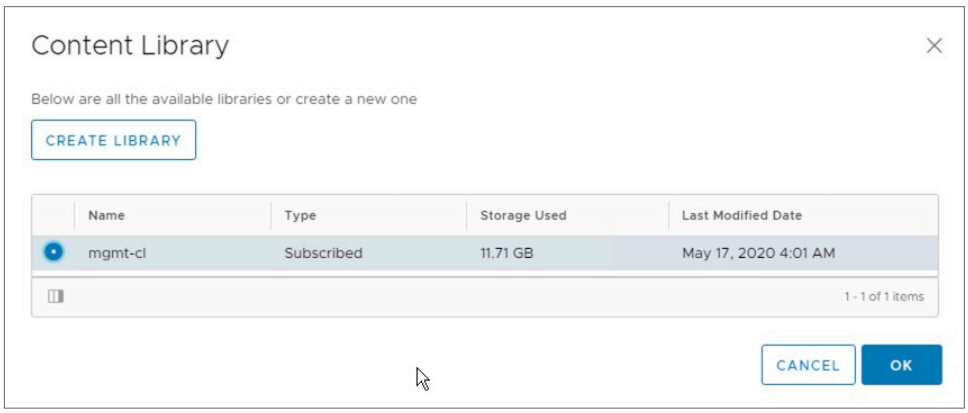
Click **OK.**



Under **Tanzu Kubernetes**, click **ADD LIBRARY.**



Select **ADD LIBRARY.**

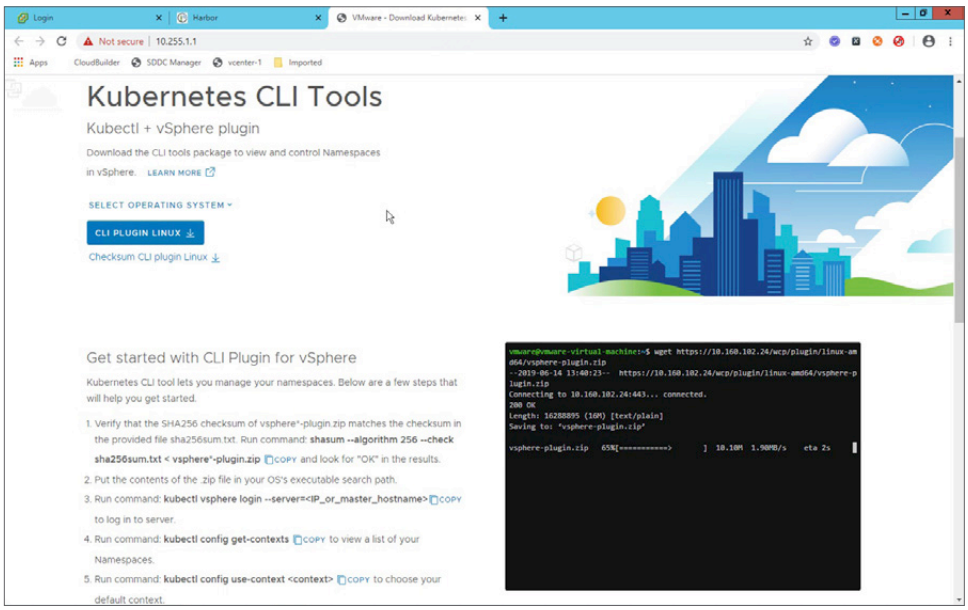


Select the **Content Library.**

Click **OK.**

This completes the steps to enable vSphere with Kubernetes on the Cloud Foundation management domain.

With vSphere with Kubernetes enabled on the management domain, we are now ready to hand off the cluster to our developers. The developers must download the **Kubernetes CLI Tools** before they can deploy workloads.



Conclusion

VMware has certified the enablement of VMware vSphere with Kubernetes on the management domain. With this change, you can now deploy the VMware Cloud Foundation consolidated architecture and enable vSphere with Kubernetes directly on the management domain.

Also with this change, you can now get started with as few as four hosts and can easily scale up to the Cloud Foundation workload domain configuration maximums.

In this paper, we provided an overview of the steps required to enable vSphere with Kubernetes on the Cloud Foundation management domain. To learn more about Cloud Foundation and to browse our library of interactive click-through demos, visit the Cloud Foundation Resource Center.

About the Author

Kyle Gleed is part of the VMware Technical Marketing team, covering VMware vSphere with Kubernetes on VMware Cloud Foundation. Kyle has been with VMware for 10 years. He spent four years working with vSphere, where he focused on VMware ESXi and VMware vCenter Server Appliance™ adoption. Over the past six years, he has specialized in the Software-Defined Data Center (SDDC), where he works closely with VMware Validated Designs and Cloud Foundation.

VMware, Inc. 3401 Hillview Avenue Palo Alto CA 94304 USA Tel 877-486-9273 Fax 650-427-5001 www.vmware.com

5/20