USING VMWARE VSPHERE VIRTUAL VOLUMES 1.0 AND VASA 2.0 WITH DELL EMC POWERMAX

A practical guide to implementing and working with vVols and the VASA 2.0 Provider

Abstract

This white paper discusses how VMware's vVols 1.0 are implemented on the VMAX® and PowerMax® using the Dell EMC VASA 2.0 Provider.

September 2020

Dell EMC Engineering

Part Number h18407

Table of Contents

Introduction

VMware vSphere Virtual Volumes (vVols) using the VASA 2.0 Provider, is available on VMAX and PowerMax arrays[1] running a minimum of HYPERMAX OS 5977 Q1 2016 (5977.811.784) or any PowerMaxOS. vVols are an integration and management framework (referred to as the vVol framework) that moves management from the datastore to the virtual machine. vVols virtualize storage, enabling a more efficient operational model that is optimized for virtualized environments and centered on the application instead of the infrastructure. vVols map virtual disks, configuration files, clones, and snapshots, directly to virtual volume objects on a storage system. This mapping allows VMware to offload more storage operations to the VMAX and PowerMax, such as VM snapshots.

vVols virtualize SAN devices by abstracting physical hardware resources into logical pools of storage called Storage Containers which replace traditional storage volumes. These containers are created on the VMAX and are assigned storage capabilities which can be all, or a subset, of the Service Level Objectives[2] the particular VMAX array supports. These storage containers are presented to the vCenter when a VASA 2.0 Provider is registered. The VASA Provider runs on the storage side and integrates with vSphere Storage Monitoring Service (SMS) and ESXi *vvold* service to manage all aspects of the storage in relation to vVols. All communication between VMware and the VASA Provider is out of band.

From the presented storage containers, the VMware Admin creates vVol datastores through the wizard, just like a traditional datastore. A vVol datastores has a one-to-one relationship with a storage container and is its logical representation on the ESXi host. A vVol datastore holds VMs and can be browsed like any datastore. Each vVol datastore then will inherit the capabilities of the container, e.g. SLOs, compression, etc. The VMware Admin creates Storage Policies that are mapped to these capabilities so when a VM is deployed, the user selects the desired capability and is presented with compatible vVol datastores in which to deploy the VM.

Unlike traditional devices, ESXi has no direct SCSI visibility to vVols on the VMAX. Instead, ESXi hosts use a front-end access point called a Protocol Endpoint. A Protocol Endpoint (PE) is a special device created on the VMAX and mapped and masked to the ESXi hosts. Each ESXi host requires a single, unique PE.[3] The ESXi host uses the PE to communicate with the volumes and the disk files the vVols encapsulate. By utilizing PEs, ESXi establishes data paths from the virtual machines (VM) to the virtual volumes.

vVols simplify the delivery of storage capabilities to individual applications by providing finer control of hardware resources and allowing the use of native array-based data services such as SnapVX at the VM level. vVols present the VMware Admin a granularity of control over VMs on shared storage that cannot be achieved

[1] This whitepaper will sometimes use the generic term VMAX when referring to VMAX3, VMAX All Flash, and PowerMax arrays. When necessary the individual array type will be called out.
[2] In more recent releases of the VMAX and the PowerMax, SLO is renamed to SL or Service Level. Both terms will be used in this whitepaper interchangeably.
[3] The restriction of only one PE per host may be lifted in a future release.

with traditional architecture due to the device limitations of the ESXi server. Dell EMC supports up to 64k vVols on a VMAX or PowerMax system. Note that as each vVol uses a host available address, each one counts toward the total number of devices on the array.

This paper will start with an explanation of how virtual volumes are managed including how the VMware storage APIs relate to vVols. There will also be a brief introduction to VASA 2.0 before embarking on the technical details of the implementation, namely, how to install, configure and use the VASA 2.0 Provider with virtual volumes in VMware vSphere 6 environments with VMAX storage arrays. An understanding of the principles that are exposed here will allow the reader to deploy and utilize virtual volumes in the most effective manner.

Minimum Support Level

Support for VMware vVols requires a minimum of HYPERMAX OS 5977 Q1 2016 release and Management Software Version 8.2 on VMAX or any PowerMaxOS release on PowerMax and Management Software Version 9.0. This paper covers up to the PowerMaxOS 5978 Q3 2020 release and the VASA 2.0 Provider version 9.2.

Audience

This technical white paper is intended for VMware administrators and VMAX administrators responsible for deploying VMware vSphere 6.x and 7.x, with VMware vVols, on VMAX or PowerMax.

Due to the nature of development, the exact minor revisions of products in this paper may not match those available to customers at the time of general availability (GA). Note, however, that every effort was made to ensure the functionality between the versions in this paper and those at GA are the same.

Virtual Volume Management

vVols involve both a storage role and a VMware role. In some companies these two roles are consolidated, but in most large enterprises these are distinct positions, and as such there is a desire for bifurcation of tasks when it comes to virtualization and storage. vVols offer that separation with the storage administrator (SA) maintaining control over the physical storage requirements as well as where that storage is made available, and how much. The VMware administrator, meanwhile, maintains the ability to create VMs and select from available service levels (SL) that the SA has presented to the storage container(s).

Storage Administrator role

The SA is responsible for three main tasks:

1. Provision and present Gatekeepers and a device for the VASA DB to the ESXi host (or cluster for VMware HA) where the VASA Provider will be deployed.
2. Provision and present PE(s) to the ESXi host(s).

3. Create storage container(s) and assign storage resources in the desired storage amounts and SLs.

These tasks can be accomplished through Unisphere for VMAX, Unisphere for PowerMax, or Solutions Enabler. Unisphere is recommended as the wizards provide an easy interface to deploy both objects. There is an additional capability available to the SA, and that is general monitoring of the VASA Provider (VP). Unisphere provides an interface to add the VP IP and once added Unisphere will monitor the viability of the VP by making direct calls to the array.

All tasks are covered in the sections Using Unisphere with Virtual Volumes and Using Solutions Enabler with Virtual Volumes.

One thing to note with vVols is that the SA has no control over replication of the disks or virtual machines. Clones and snapshots that utilize TimeFinder technology can only be accessed through the vSphere interface. It is not possible, for instance, to use Unisphere or Solutions Enabler to copy a vVol device.

VMware Administrator role

Once storage is presented to the ESXi hosts (GKs, VASA DB device, PE), the VMware administrator is then able to proceed with his/her tasks:

1. Deploy the VASA Provider including mapping Gatekeepers/VASA DB device and mounting the VASA database.
2. Register the VASA Provider in VMware vSphere vCenter.
3. Create vVol datastores from presented storage container(s).
4. Create Storage Policies for each service level supported and advertised by the system.
5. Create virtual machines and execute clones and snapshots of those virtual machines.

The VMware admin manages the lifecycle of virtual machines, including snapshots, clones, fast clones, etc. through the vSphere Web Client[4]. When the VMware admin takes a VM snapshot, TimeFinder SnapVX[5] technology is utilized so the snapshot is far more efficient than the traditional VMware implementation. VMware no longer has to keep track of multiple delta files which can grow beyond the original size of the VM and impact performance. Each snapshot will create no more than a single vVol (only if memory is included, otherwise the snapshot is targetless) and is maintained by Dell EMC on the VMAX.

[4] The vSphere Client (thick) can still be used to accomplish some tasks in vSphere 6.0, however new capabilities related to vVols are only available through the Web Client or vSphere Client (HTML5), such as creating the vVol datastores.
[5] TimeFinder must be licensed on the array or vVol snapshot creation will fail.

Replication

vVols 1.0/VASA 2.0

An important thing to note about vVols 1.0 and VASA 2.0 is that it has no support for remote replication. There is no Site Recovery Manager support nor any vendor replication such as SRDF. That support was added in vVols 2.0 and VASA 3.0. Dell EMC, however, supports local replication through VM snapshots in vVols 1.0, which utilize the underlying TimeFinder SnapVX software, taking full advantage of the performance benefits of that technology. This means no additional virtual volumes are required when taking a VM snapshot, unless including the machine's virtual memory which will produce one additional virtual volume. When a snapshot is restored, the VMAX creates and then deletes the necessary virtual volumes for the process.

Dell EMC currently limits the number of snapshots to 12, not including the source. VMware's limit is 31, not including the source. If the user attempts to take more than 12 snapshots, the error in Figure 1 will be generated.

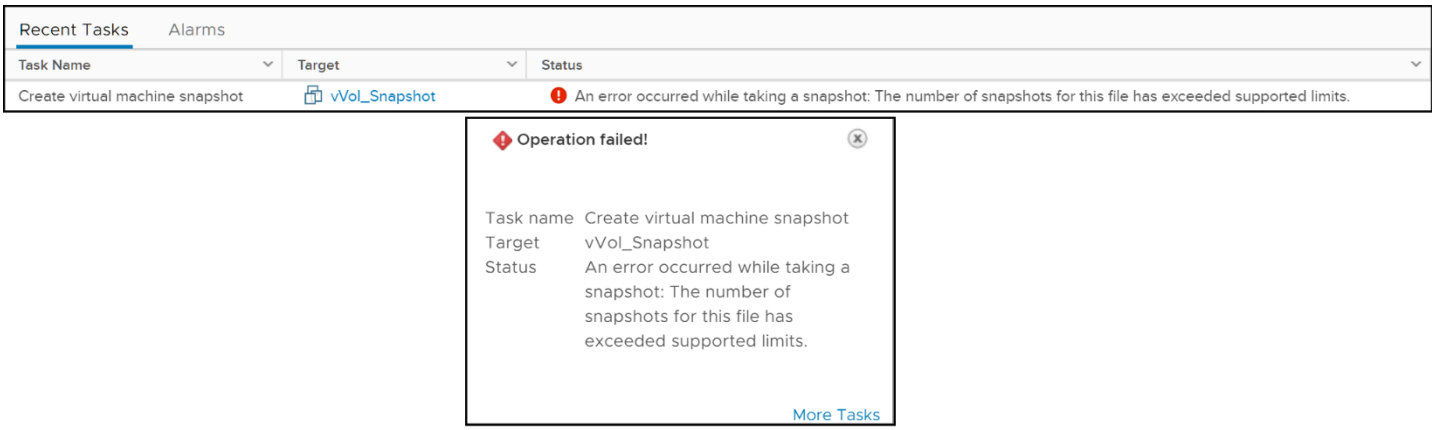

Figure 1. Error upon taking too many snapshots

While it is not possible to adjust this value directly, if the 12-snapshot restriction is deleterious to the business, an SR can be opened with Dell EMC Support and a request made to increase it.

vVols 2.0/VASA 3.0

With vSphere 6.5, VMware released vVols 2.0 which supports array remote replication. Dell EMC supports vVols 2.0 and remote replication (SRDF/A) with the embedded VASA 3.0 Provider version 9.2 (eVASA) and vSphere 7.0 only. The embedded provider is only available for order on new PowerMax arrays running PowerMaxOS 5978 Q3 2020. For more information on implementing vVols 2.0 and VASA 3.0, please see the white paper *Using VMware vSphere Virtual Volumes 2.0 and VASA 3.0 with Dell EMC PowerMax*.

The Dell EMC VASA 2.0/3.0 Providers do not currently support SCSI-3 reservations.

Compression/Deduplication/Data Reduction

Virtual Volumes support compression on the VMAX All Flash and data reduction (compression and deduplication) on the PowerMax arrays (PowerMax 2000 and 8000). Compression and/or data reduction is enabled at the storage resource level as in Figure 2.

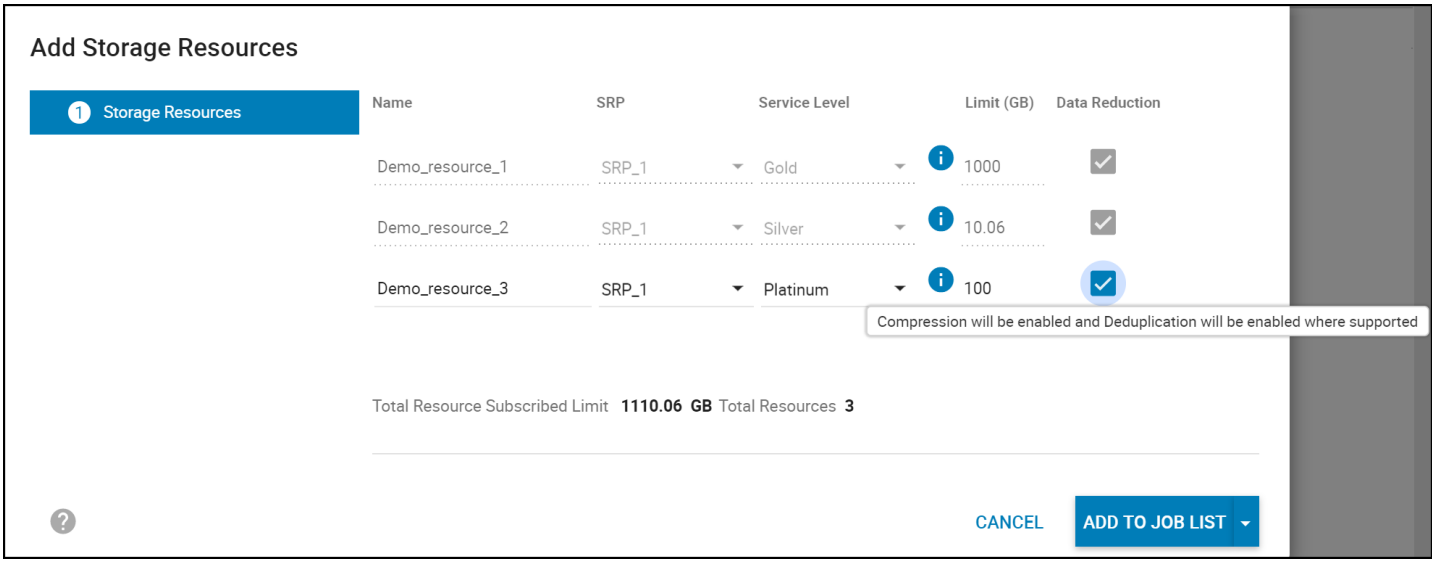

Figure 2. Enabling compression/deduplication on storage resource

Because deduplication cannot be separated from compression, there is only a single check box no matter what array is in use with vVols. While compression is only applicable to data within a storage resource where it is active, deduplication applies to all data in an SRP when that data is in a storage resource or storage group with compression/deduplication active. The following paragraph provides a visual example of how this feature works.

A single virtual machine, VM _1, is created in storage resource A with compression/deduplication active. Its OS vmdk (vVol) is 20 GB and identified as D6. The array recognizes that many tracks can be compressed and separates them into smaller extent pools. These are named DG1_F_x in Figure 3.

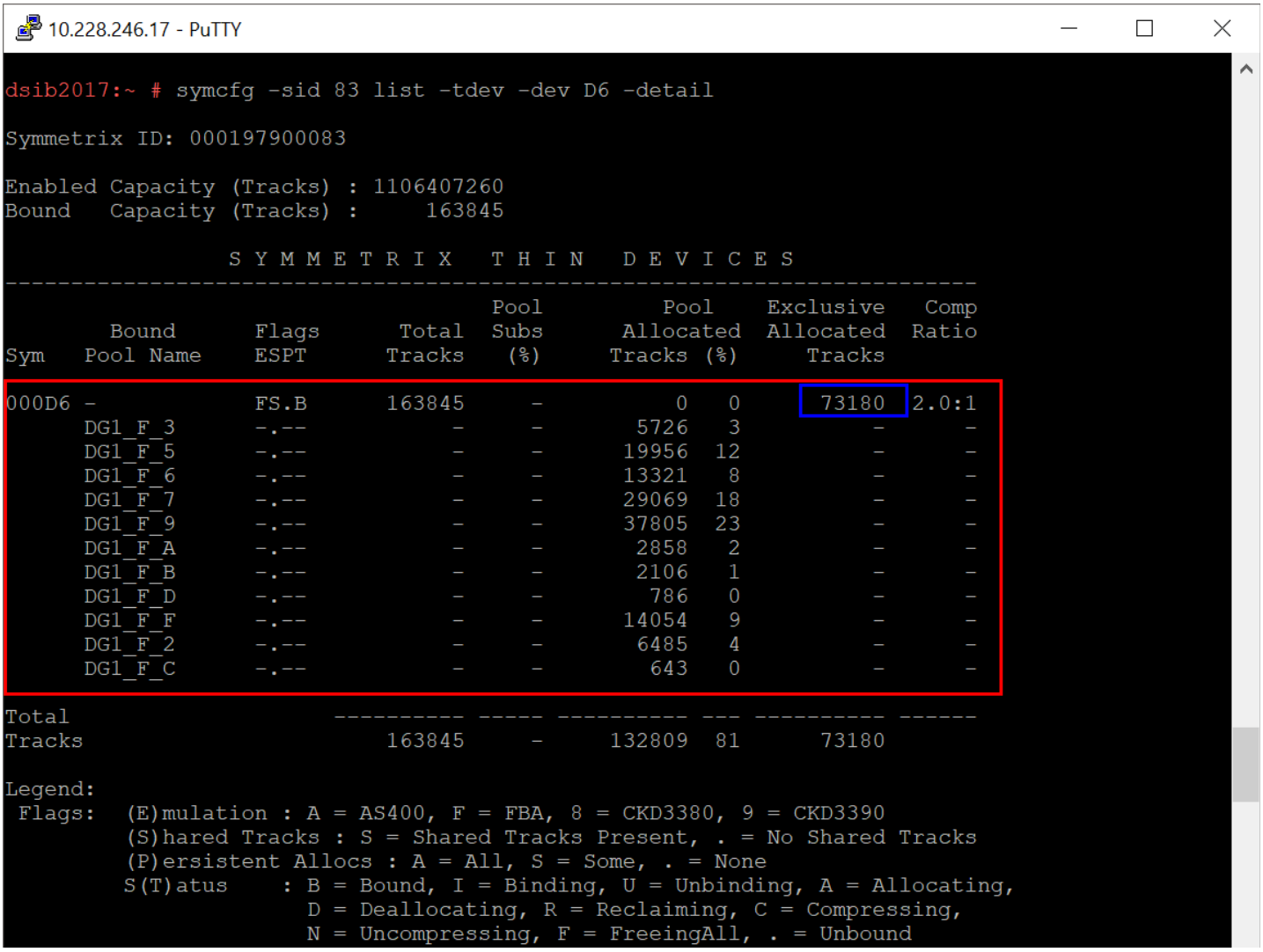

Figure 3. Compressed/Deduplicated tracks in a vVol

In addition, the array is also able to remove tracks through deduplication. The total number of tracks allocated in a pool is 132809, however in the blue box in Figure 3, the number of tracks that are exclusive to this device are only 73180. This means that 59629 tracks are shared with other devices that have deduplication active in the SRP. Note that it does not matter whether those devices are vVols or regular TDEVs or even in use by VMware.

Now, suppose VM_1 is cloned into storage resource B with compression/deduplication active and that new vVol device in VM_2 is identified as D8. If the tracks are viewed for this new VM, note how in Figure 4 there are no longer any exclusive tracks.

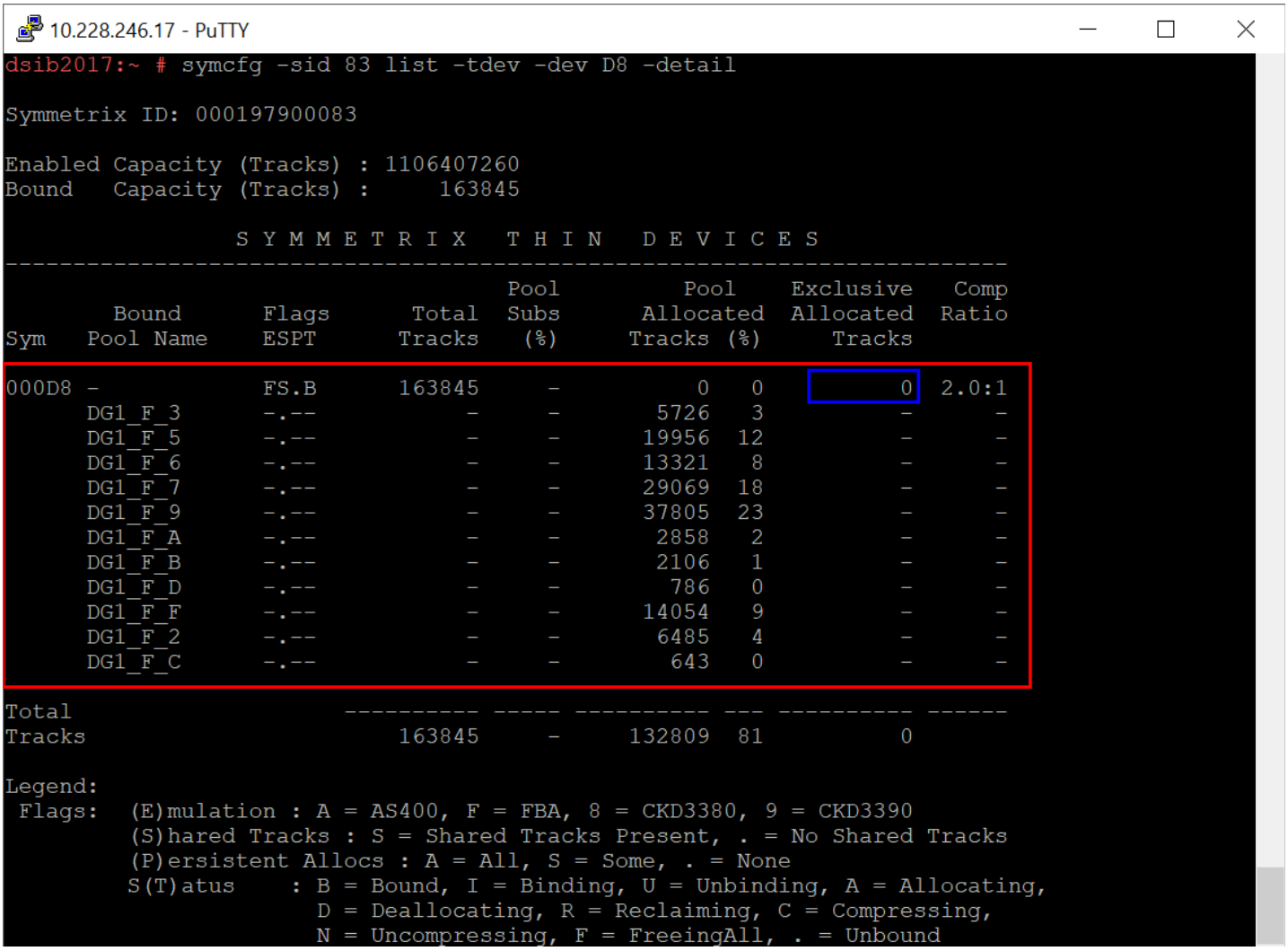

Figure 4. Compressed/Deduplicated tracks in a cloned vVol

All tracks are now shared with D6. In fact, if D6 is viewed after this clone, all its exclusive tracks are gone for it because it has the reciprocal relationship with D8 (Figure 5).

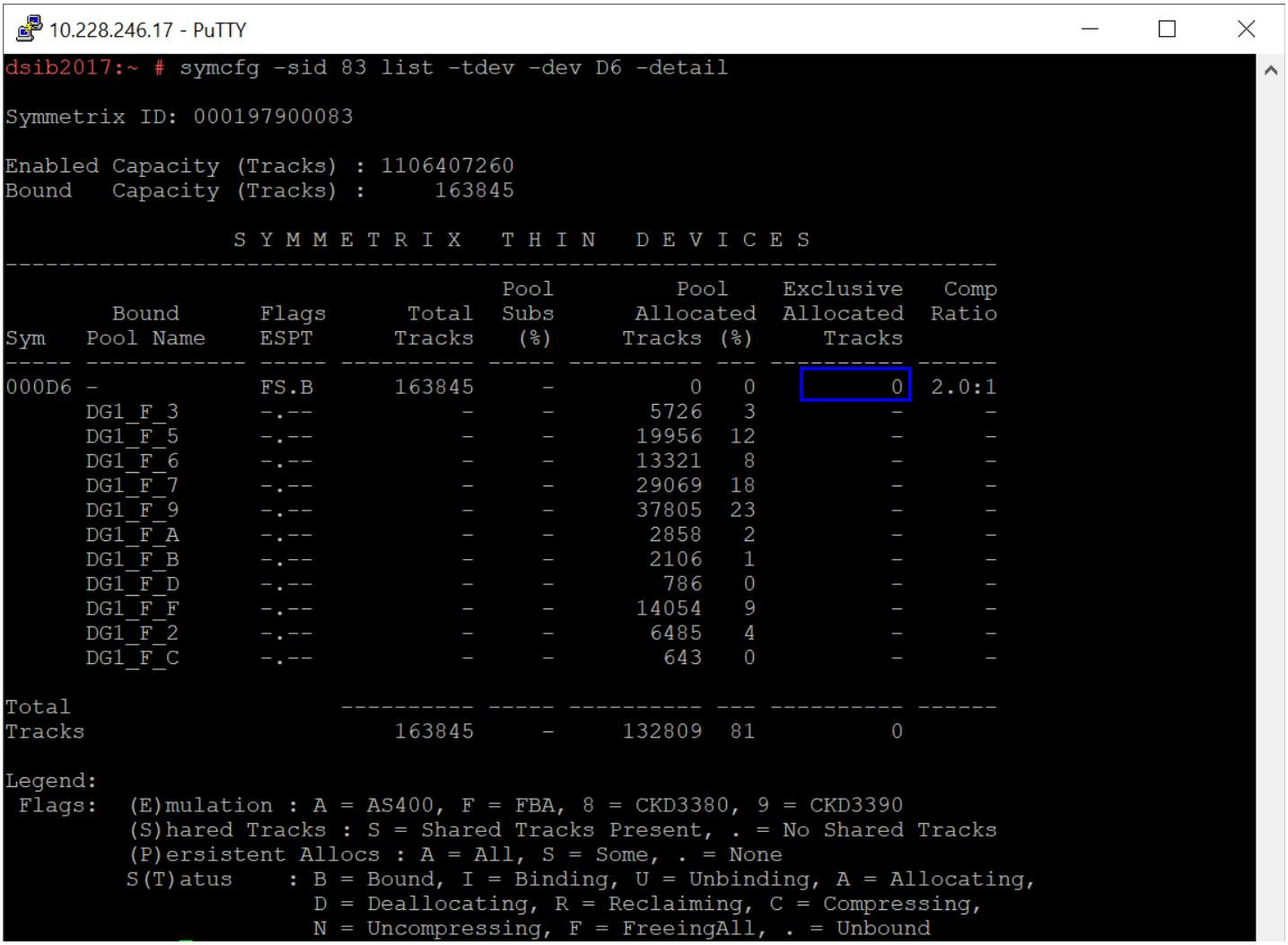

Figure 5. Removal of exclusive tracks post clone

Virtual Storage Integrator

There is no version of VSI that supports vVols. If VSI is installed in the vSphere Client, for instance, VSI will be unable to resolve any information about the datastore and will produce the standard message in Figure 6: *This datastore is not on a registered Dell EMC storage system.*

Currently, from within vCenter, it is not possible to map a vVol vmdk back to the underlying storage device, however this can be done in Unisphere for PowerMax and is covered later in the paper.

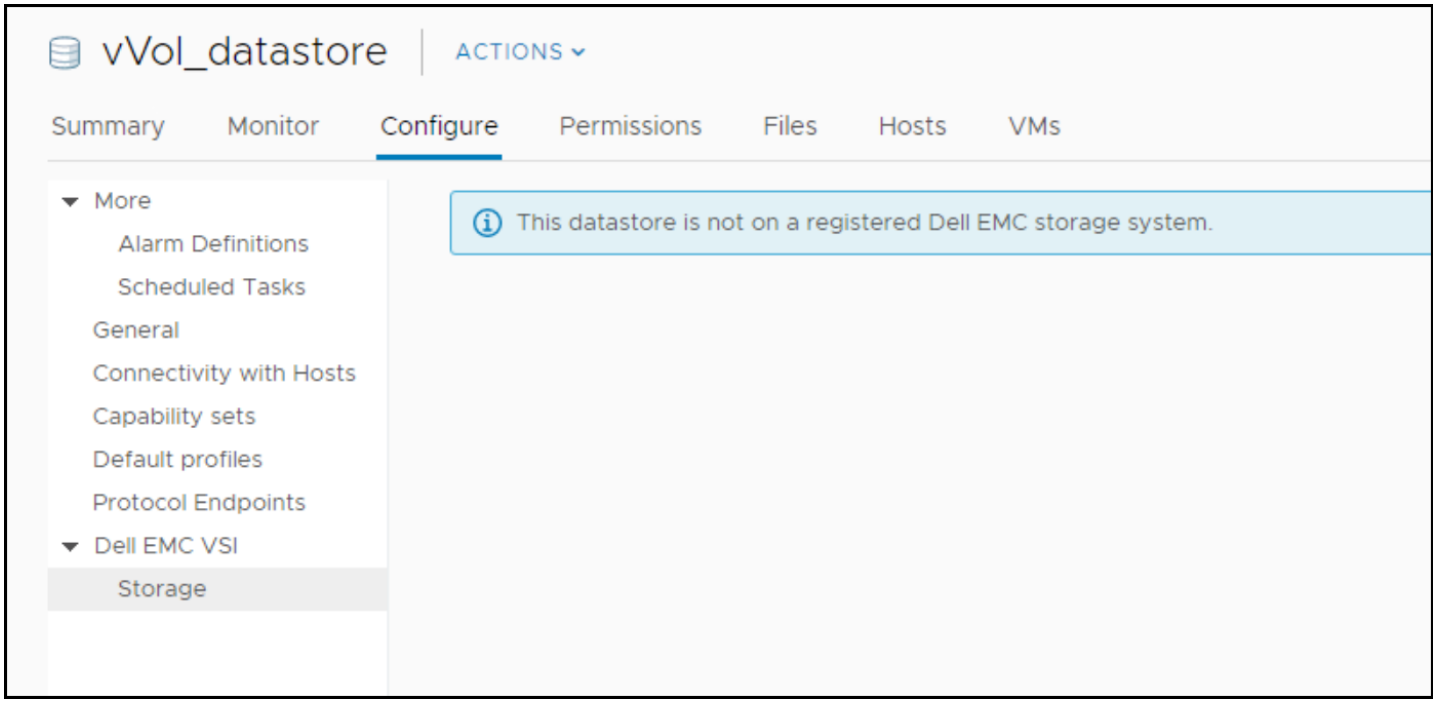

Figure 6. VSI with vVols

VAAI and vVols

With vVols, there is less reliance on VAAI primitives, either because they are not needed or the VASA APIs are more efficient and used instead; however, VAAI and thin-provisioning primitives do co-exist with vVols.

For vVols, ESXi will generally try to use the VASA API primitives as the default behavior. If these are not supported, it will fall back to software.

This paper will not cover the VAAI primitives in detail, rather it will focus on when they might be used in a vVol environment. For an in-depth look at the VAAI primitives see the following whitepaper: *Using VMware vSphere Storage APIs for Array Integration with Dell EMC VMAX and PowerMax*.

ATS

All config vVols are formatted with VMFS and hence require supporting ATS commands. This support is detected based on ATS support for a PE LUN to which vVols are bound.

Full Copy (XCOPY)

In vVols 1.0 on the VMAX with vSphere 6.x and 7.x, VMware is unable to send XCOPY commands to vVols. Architectural changes are being implemented in a future release of vSphere and PowerMax, at which point XCOPY will be supported.

Block Zero (WRITE SAME)

Block Zero is most commonly used to zero out space on VMFS – in particular for eagerzeroedthick disks. As vSphere 6.5, 6.7 and 7.0 do not use eagerzeroedthick for vVols there is little need of this primitive.

Reclaim (UNMAP)

In traditional VMFS, the *unmap* command can be issued at the datastore level (manual or auto) on the ESXi host to reclaim space vacated by disk deletion or Storage vMotion. Since vVols are basically individual vmdk files, it is not possible to issue UNMAP against the vVol datastore. If an attempt is made to run reclaim on a vVol datastore, the error in Figure 7 will be returned, indicating the datastore is not of the VMFS type.

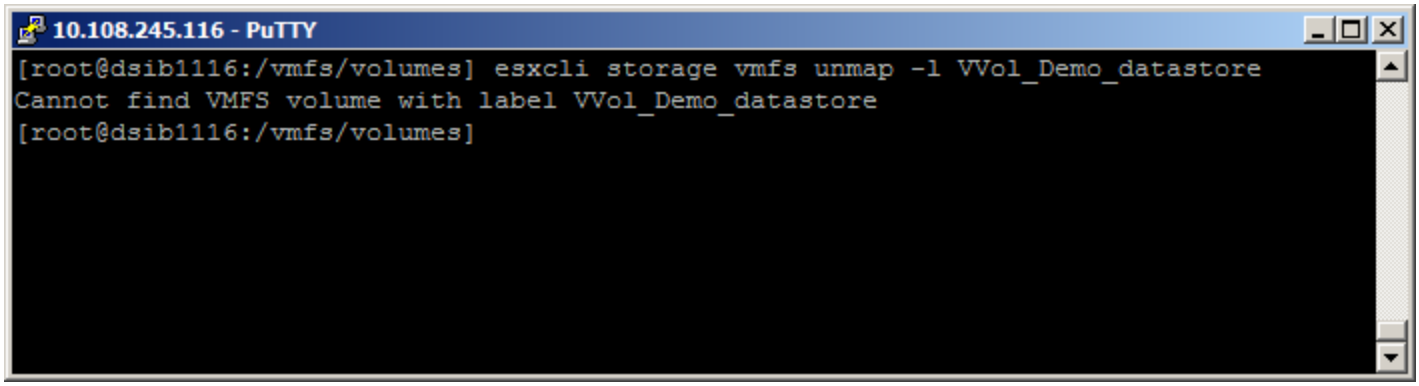

Figure 7. UNMAP attempt on vVol datastore

The fact that UNMAP does not work on vVol datastores is of no great concern. This is because a vVol is both a vmdk and a device on the VMAX array. When you delete a vVol, therefore, the VMAX device (TDEV) is also deleted and the space it once occupied is immediately freed from the Storage Resource Pool (SRP).

Because vVols are vmdks and by default of thin type, they do support UNMAP at the Guest OS level. There are some restrictions/prerequisites that must be met on the VM and the Guest OS. These can be found in VMware KB 2112333. This functionality ensures that no space on the array is wasted.

Thin Provisioning Out of Space

Storage container “out of space” warnings will be advertised to vSphere.

VASA 2.0 Installation and Configuration

The VASA 2.0 Provider is supported for installation only as a virtual appliance as detailed in the software’s release notes. Both the software and documentation can be downloaded from support.emc.com and the release notes contain a complete list of all system requirements along with known limitations. This whitepaper is not a substitute for the product documentation, rather it can be used in addition to it.

The SMI-S Provider has not been included in the virtual appliance which contains the VASA 2.0 Provider. If the SMI-S Provider is required, a separate installation of Solutions Enabler must be undertaken which bundles the SMI-S Provider.

Installation

The virtual appliance for the VASA 2.0 Provider (VP) can be deployed using the vSphere thick client (6.0) or vSphere Web Client/vSphere Client (HTML5) on vSphere vCenter 6.x. It is not required that the vCenter be the same one which will be used with vVols; however, the ESXi host on which the VP runs must be zoned to the vVol array and be presented with Gatekeepers (GK). It is important, too, that the VP clock be in sync with the vVol vCenter clock.

The VP requires at least 5 GKs and a single 4 GB device for use as the VASA database. Although it is possible to add the GKs and VASA device as physical RDMs prior to powering on the VP, the recommendation is to use the GUI configuration screen of the vApp to add them to the VM as there is no other method to mount the VASA DB. This means the devices need only be presented to the ESXi host(s) where the vApp is deployed as the GUI configuration process will map them as RDMs.

Dell EMC recommends that the VASA 2.0 Provider is not installed on the same ESXi host (or cluster) as any vCenter Server Appliance (VCSA), as the VCSA can require all system resources. Contention for resources can produce unintended side effects to the VP.

In large scale vVol environments, it is possible to experience VMware task timeout failures when running multiple, simultaneous activities on vVol VMs (e.g. vMotion), particularly when the VMs have multiple snapshots. If these events occur, Dell EMC recommends increasing the Gatekeeper count from 5 to 15.

High Availability

The VASA 2.0 Provider is not highly available by default. As it runs as a vApp external to the array, however, it is possible and recommended to use VMware HA. In the event of a down ESXi host the VP will restart on another host in the cluster.

Disaster Recovery

There is no disaster recovery solution for the VP, e.g. RecoverPoint, SRDF, because the VASA DB cannot be copied (local or remote). In the event of a failure of the VP itself, please consult the section on VASA Provider Recovery.

Backup

It is not possible to back up the VP because the VASA database cannot be backed up. Loss/corruption of the vApp should be resolved by redeploying the VP. Please consult the section on VASA Provider Recovery for instructions.

Deployment

The VP is provided as an OVA file. The following section details only the critical aspects of the deployment. The assumption is that the user is familiar with deploying an OVA file and therefore only the customization portion is included. The deployment in this example uses release 9.2 of the VASA Provider and the vSphere 7.0 Client (HTML 5). Note that despite different versions of the VASA Provider available on support.emc.com, e.g. 8.2, 8.4, each one contains the VASA 2.0 Provider.

Figure 8 shows the customization template screen where values such as the DNS are specified. It is essential when deploying the VP that the DNS information is correct. VP will use the IP of the vApp along with the DNS to determine the FQDN. Be sure a forward and reverse *nslookup* of the IP using the DNS will return the correct FQDN. See the Appendix: VASA Provider/Virtual Volume Troubleshooting for more information.

Because of known issues, do not use the word “local” in any server or DNS naming.

DO not rename the VASA Provider host or change the IP address once deployed. Doing so may cause registration of the VP in the vCenter to fail. If the IP or hostname must be changed, Dell EMC recommends redeploying the VP and retry registration if possible.

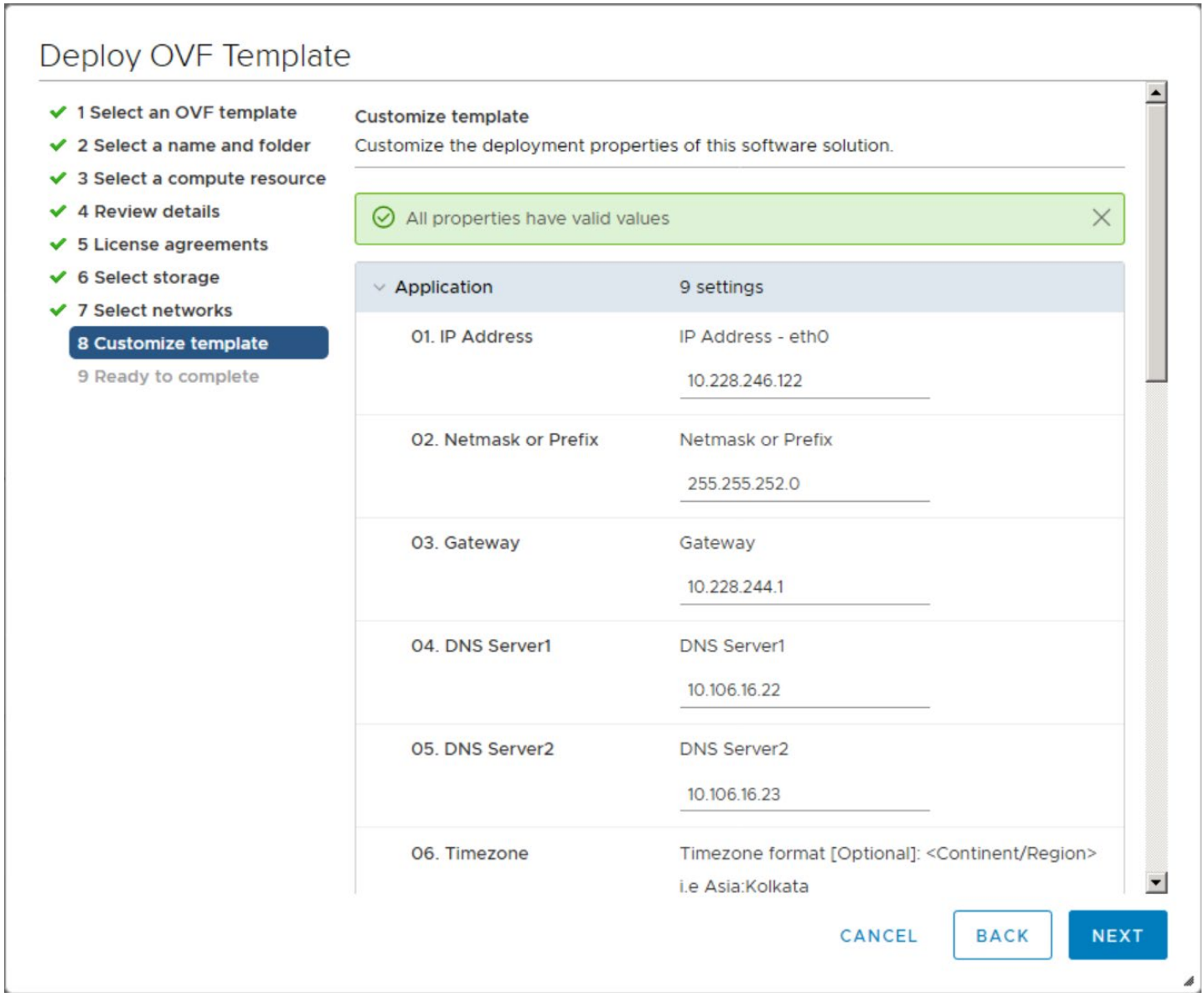

Figure 8. OVF deployment parameters

Once deployment is complete and the VM powered on, navigate to *https://<FQDN>:5480* which is the login screen for the VP configurator seen in Figure 9:

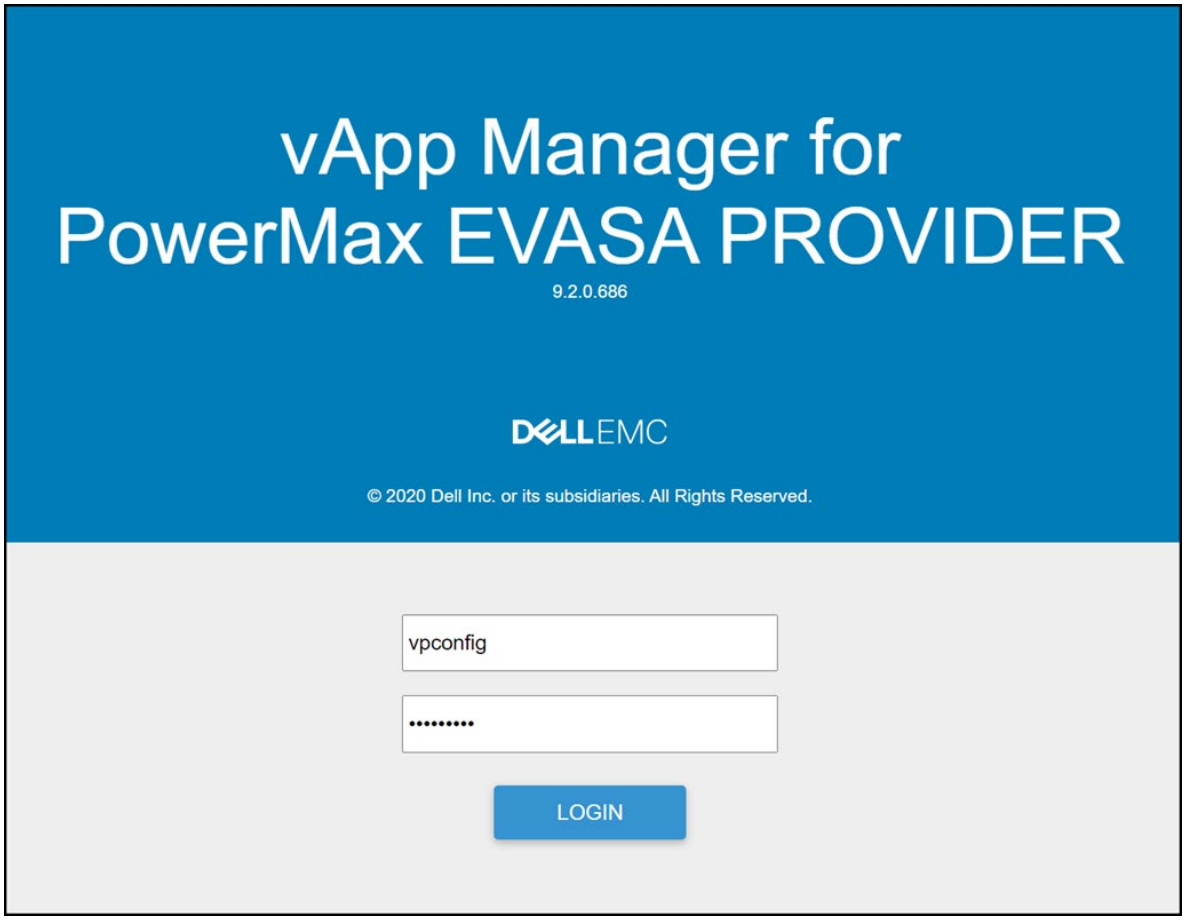

Figure 9. VP vApp login

The initial login is *vpconfig/vpconfig* but a password change will immediately be requested as in Figure 10.

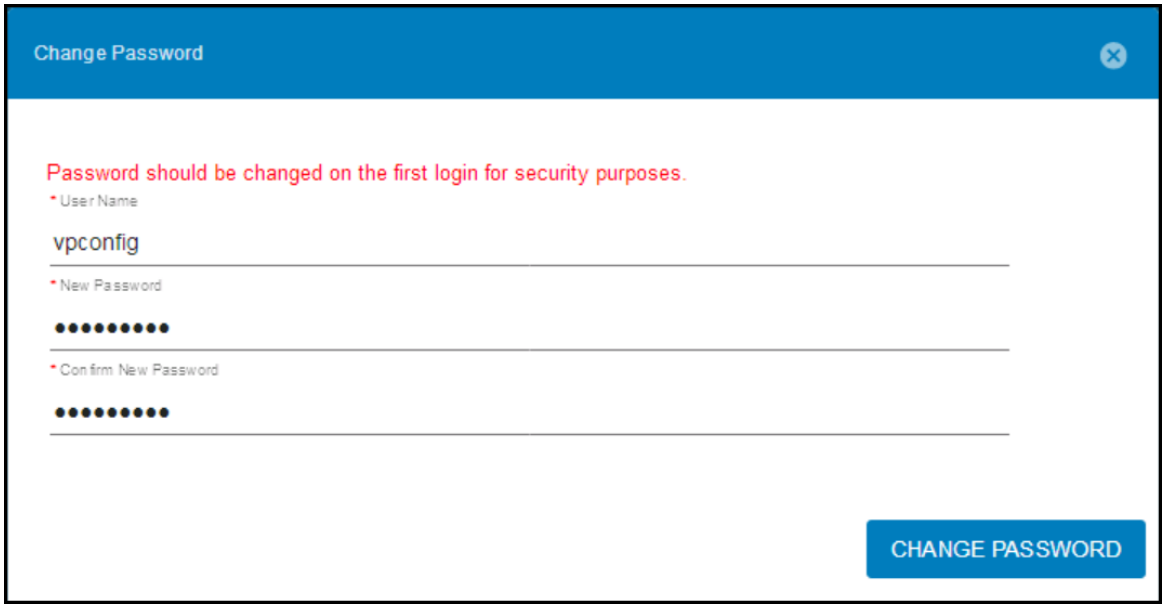

Figure 10. VP vApp password change

Upon login, the dashboard presents the system information, network, Symmetrix arrays (if any), authority status, and disk usage (Figure 11):

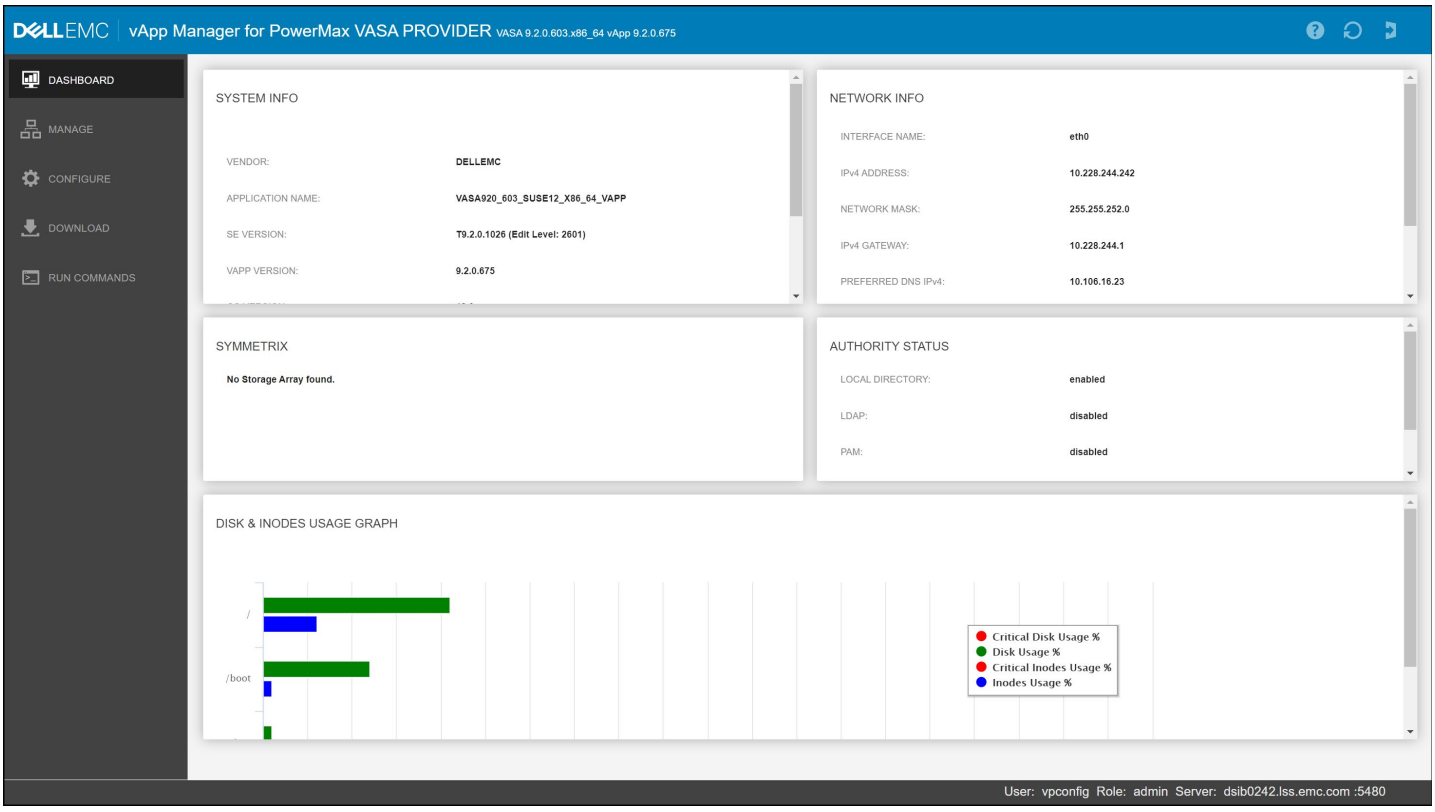

Figure 11. VP vApp dashboard

Even if GKs were mapped before powering on the VM, start by navigating to the GATEKEEPERS menu on the left-hand panel MANAGE -› GATEKEEPERS. Here,

displayed in steps in Figure 12, start by adding the FQDN of the ESXi host where the VM is running. Although the interface may allow the use of the IP address here, it is not supported.

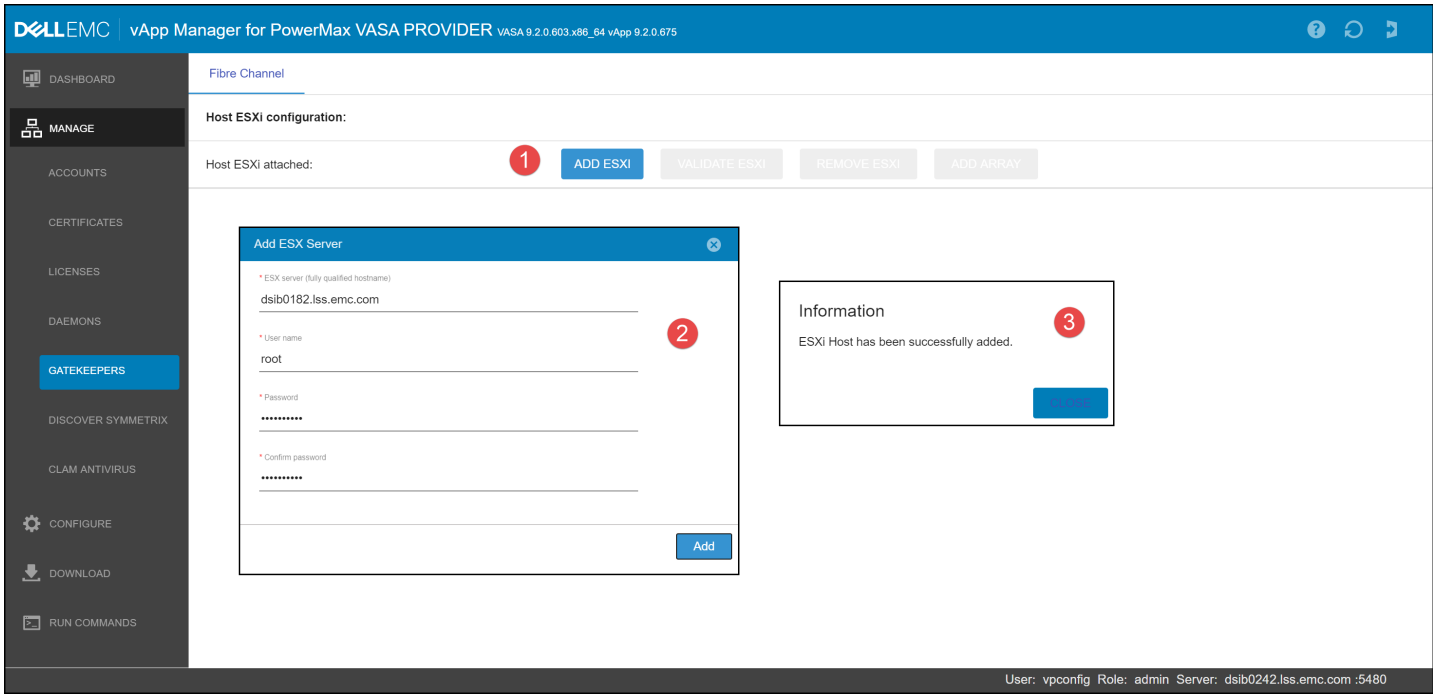

Figure 12. Adding the ESXi host in the VP vApp

Once the ESXi host is added, the VALIDATE ESXI button will no longer be grayed-out. Select it and validate the certificate as shown in steps in Figure 13.

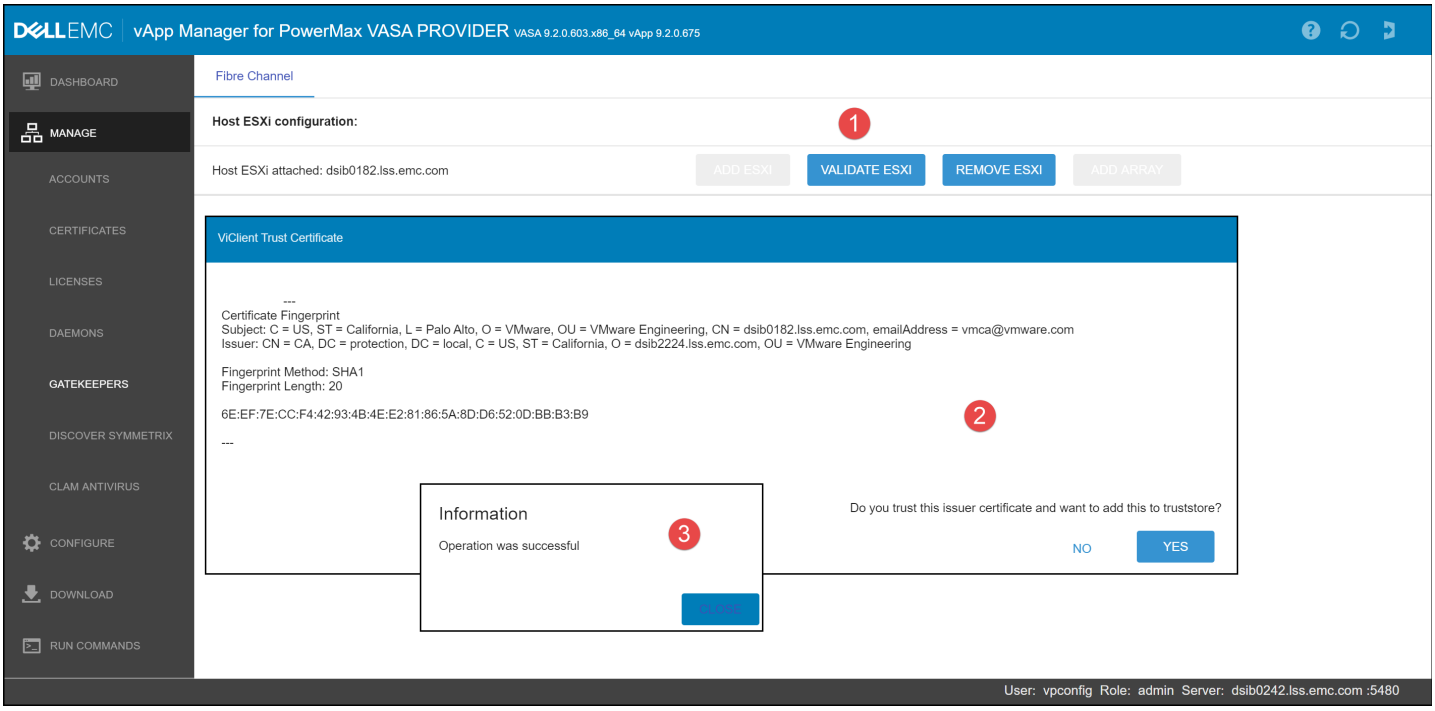

Figure 13. Validating ESXi host

Once the ESXi host is validated, the ADD ARRAY button will no longer be grayed-out. Select it and add the appropriate array as shown in steps in Figure 14. Note that all arrays that have storage presented to the ESXi host will appear, so be sure the correct array is selected or the correct devices for mapping will not display. The VP only supports a single array so do not add GKs from any other array once an array is selected and GKs mapped.

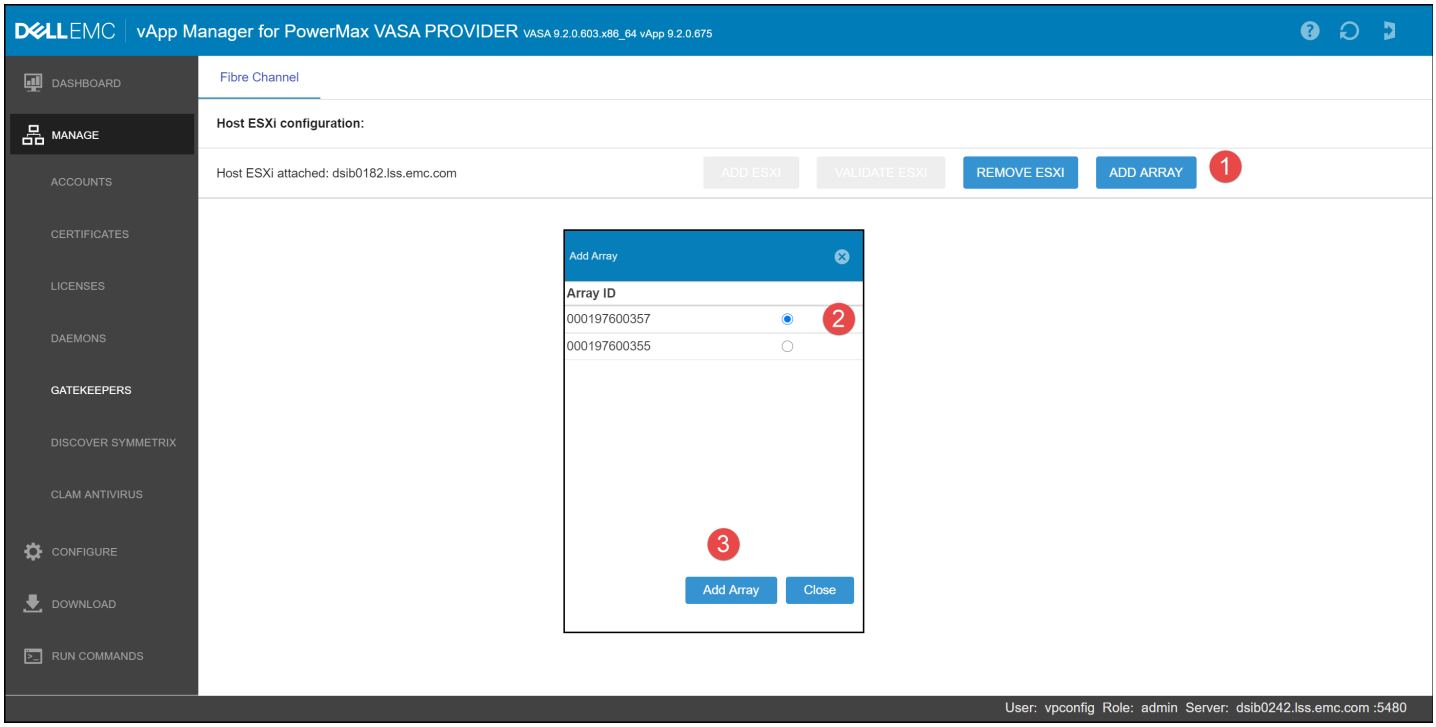

Figure 14. Adding the array in the VP vApp

If the array list shows no arrays, or the word “found” is shown, there is a DNS issue. Follow KB article 000531448 to resolve.

With the array now added, all the devices presented to the ESXi host from the array will appear in the screen in Figure 15.

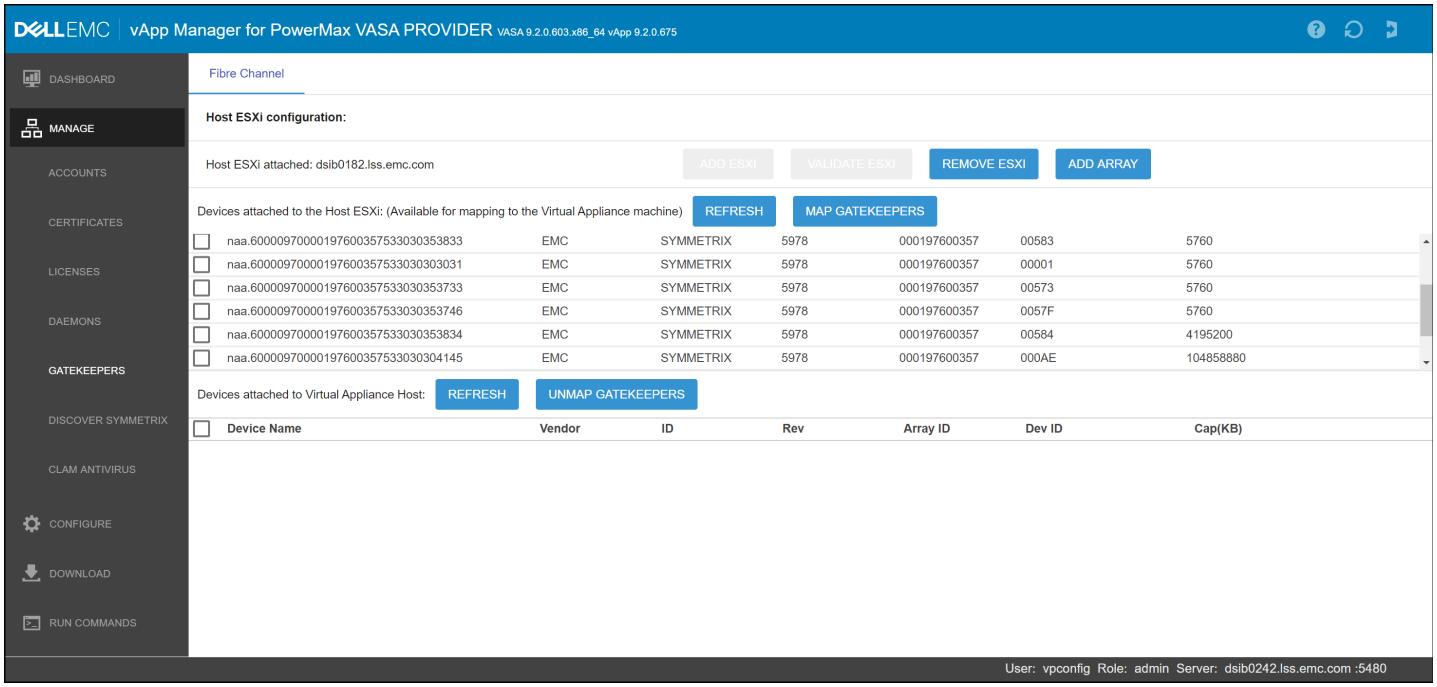

Figure 15. Devices available for mapping in the VP vApp

Using the checkboxes on the left-hand side, select at least 5 GKs for mapping and the VASA database device. Note once the devices are mapped, the MountDB button will only show for those devices large enough to support the VASA DB. Figure 16 is an example of 6 GKs and the 4 GB VASA DB presented to the VM. Note that if the devices were mapped as RDMs prior to powering on the VM, they will be shown here once the ESXi host and array are added.

Navigating away from the GATEKEEPERS screen will require re-adding the array to see the devices both mapped and unmapped when the tab is selected again; however, the ESXi host will remain. If the vApp has migrated to another ESXi host (vMotion/VMware HA), simply remove the existing ESXi host and add the one on which the vApp is currently running. Under normal circumstances, the GATEKEEPERS screen is only accessed during the initial configuration.

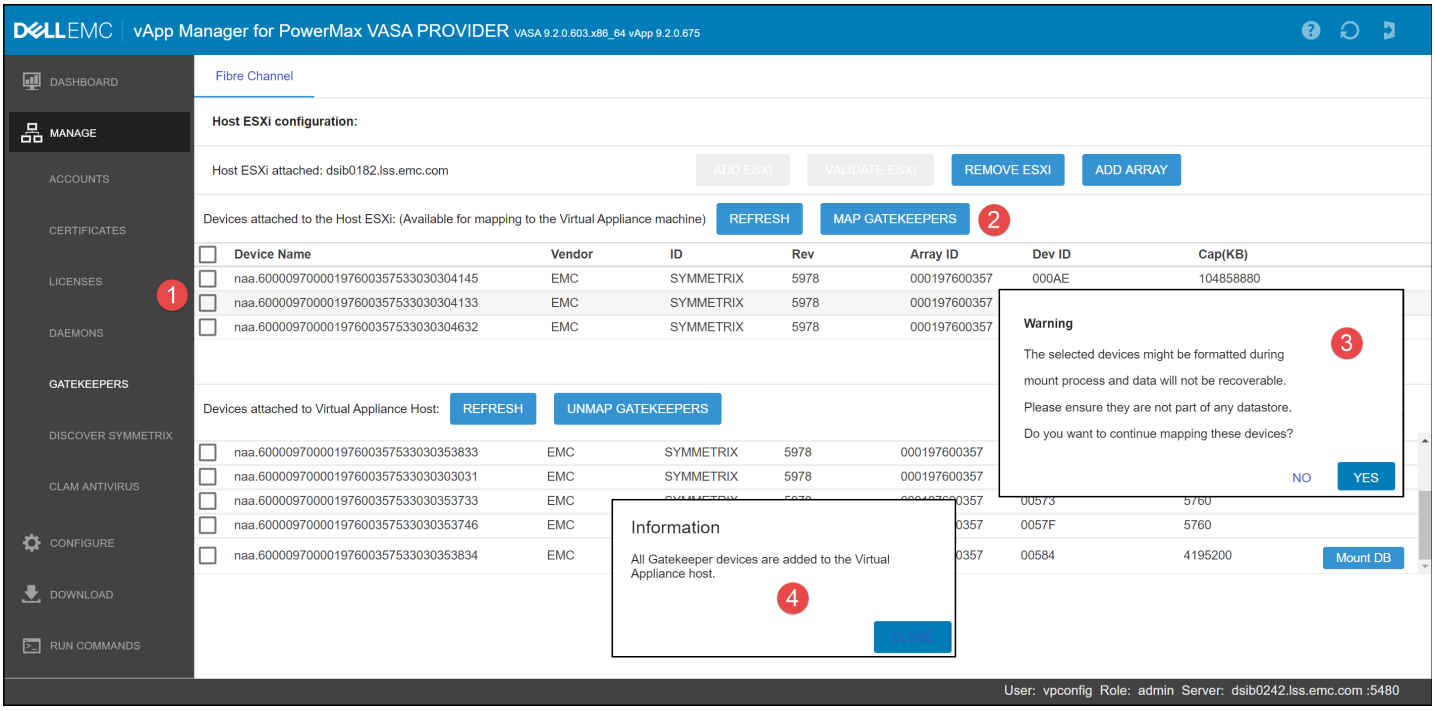

Figure 16. Mapped GKs and VASA DB in VP vApp

Now select the MountDB function for the appropriate device. One of two warnings will appear. The first depicted in Figure 17, will tell the user the device will be formatted as it is new.

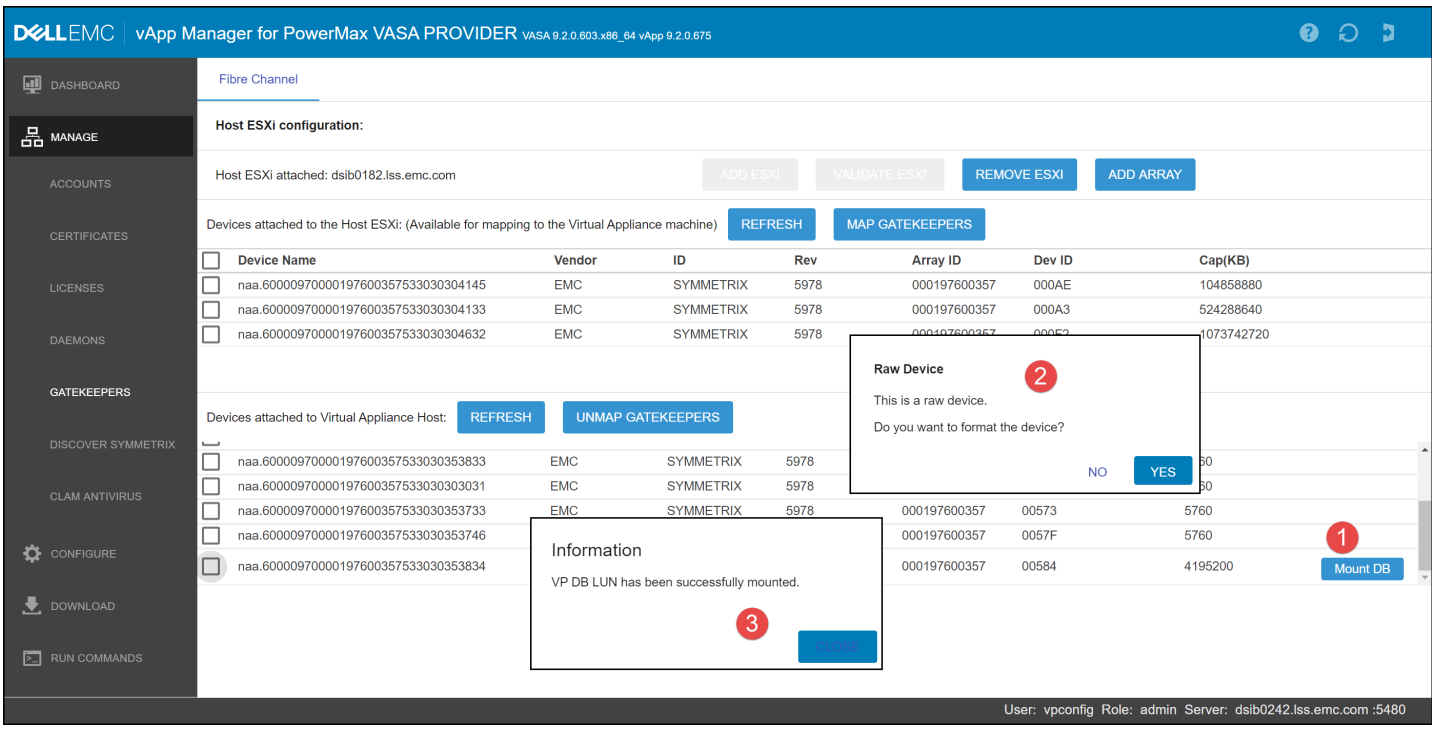

Figure 17. VASA DB LUN formatting warning

If, however, a VASA DB already exists on the device (see VASA Provider Recovery), the message in Figure 18 will appear, giving the user the option to format the device or mount the existing database.

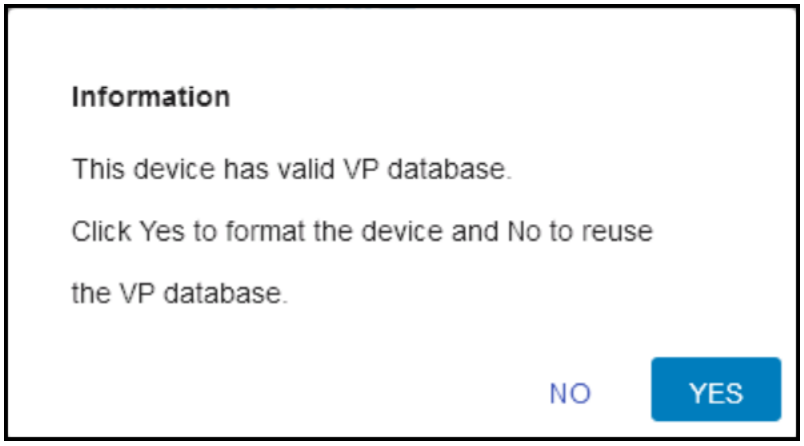

Figure 18. VASA DB LUN reuse warning

In either case, the vApp will come back with the dialog in Figure 19 when complete.

Figure 19. VASA DB mounted

It is important to note that once the database is mounted, VP will continue to offer the MountDB option next to the VASA DB device. If the user selects the button again, the following message in Figure 20 will be returned.

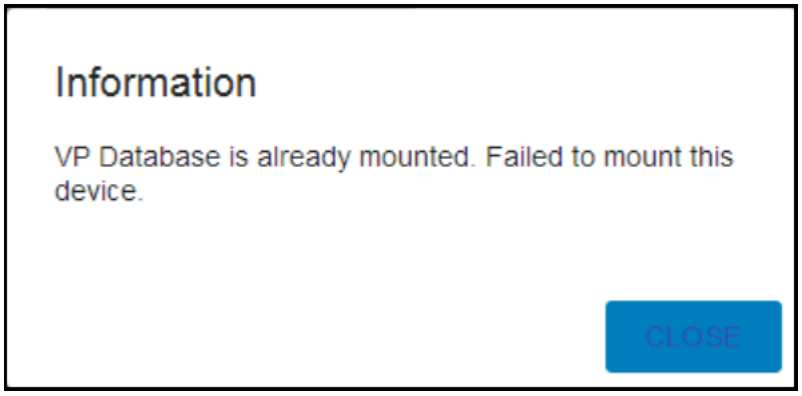

Figure 20. Re-mount VASA DB

When the database is mounted, the ECOM process, located in MANAGE-› DAEMONS, starts (Figure 21) so that can be used as a good indicator of success.

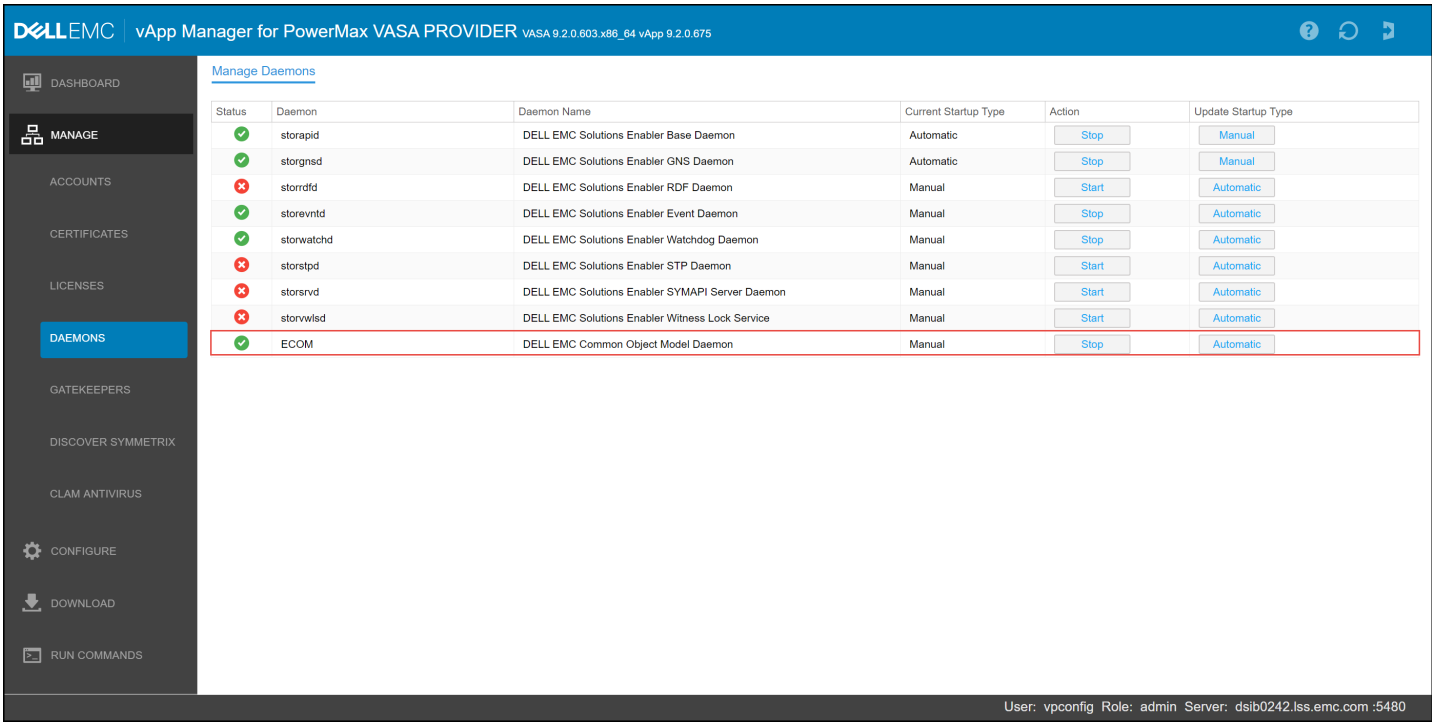

Figure 21. Manage Daemons tab in VP

If there is an issue mounting the database, the user will be directed to examine the logs. The VASA logs can be downloaded and reviewed by navigating to the DOWNLOAD/Daemon Logs screen and selecting the *VASA* Daemon as in Figure 22.

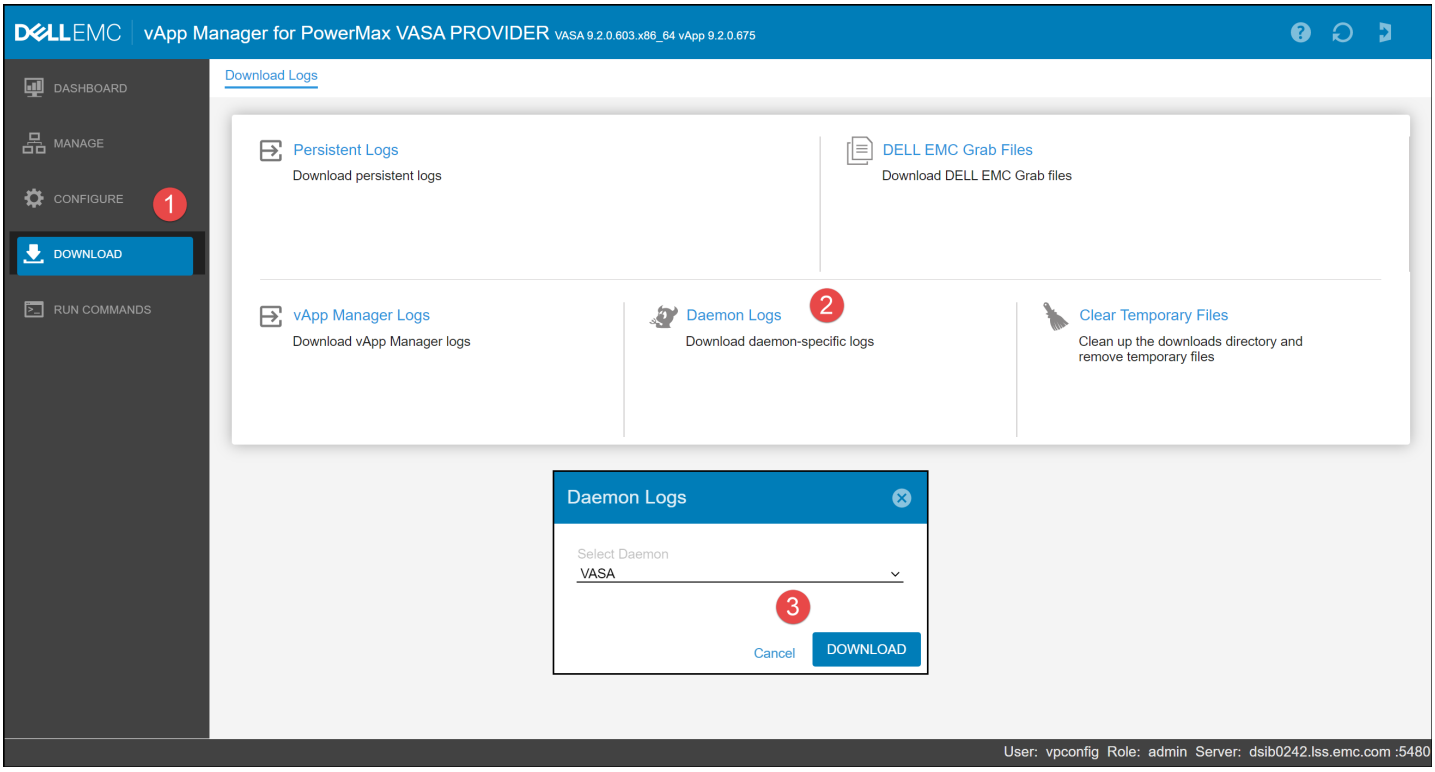

Figure 22. Downloading VASA logs

In the downloaded zip bundle, there is an “udb.log” file. At the end of this file, there should be this message if the VASA DB mounting is successful (Figure 23). Otherwise reviewing the file will reveal the issue during mounting.

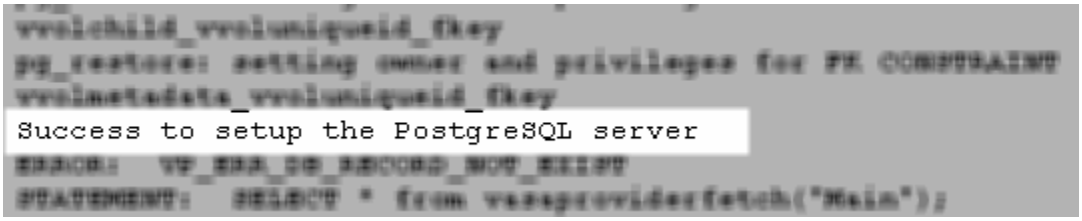

Figure 23. udb.log

Though the vApp offers some parameters to change the VP configuration in the VP CONFIGURATION screen off the CONFIGURE menu, seen in Figure 24, these should be left at their default settings unless otherwise instructed by Dell EMC Support, or if using multiple vCenters.

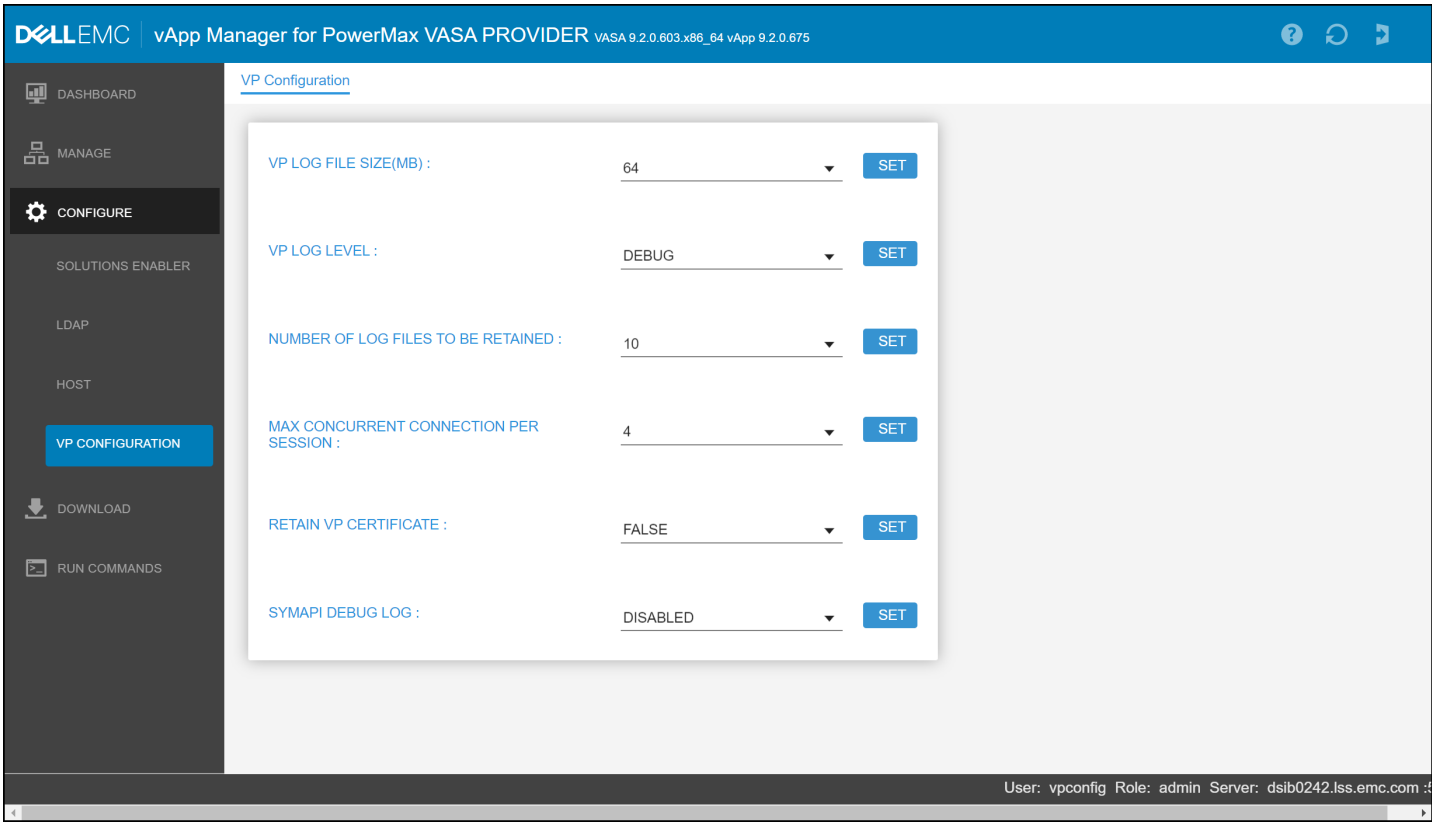

Figure 24. VP Configuration parameters

Configuration

Before the VASA Provider can be used with vCenter a few important configuration changes should be made:

1. Change the default admin password
2. Create a new user account for VASA 2.0 (admin privilege)

Configuring user authentication

It is strongly recommended to change the default administrative password (if not already changed) and to create a separate user account for vCenter access to the VASA 2.0 Provider.

The VASA 2.0 Provider offers a web-based interface for creating and managing user accounts. The default username and password as well as the URL for access are listed below:

Username: admin

Password: #1Password

Management URL: https://‹FQDN or IP of the VASA Provider›:5989/ecomconfig

The logon page for the web-based management interface is shown in Figure 25.

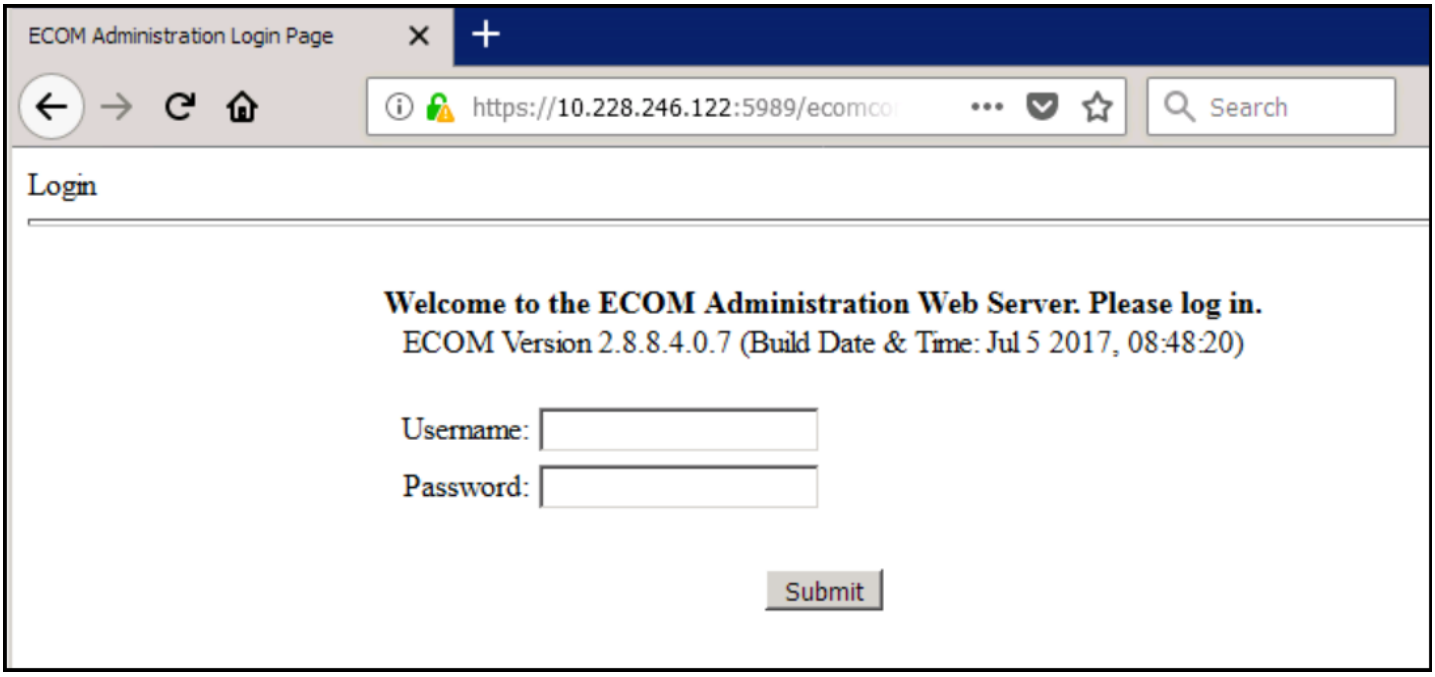

Figure 25. Logon page for ECOM web-based management

The first thing that the user should do is to change the default admin password from #1Password to a unique and complex password to prevent unauthorized access to the VASA 2.0 Provider. This process is shown in steps in Figure 26. Note that all screens are shown together, though when changing the password, the menu on the left disappears once an option is selected.

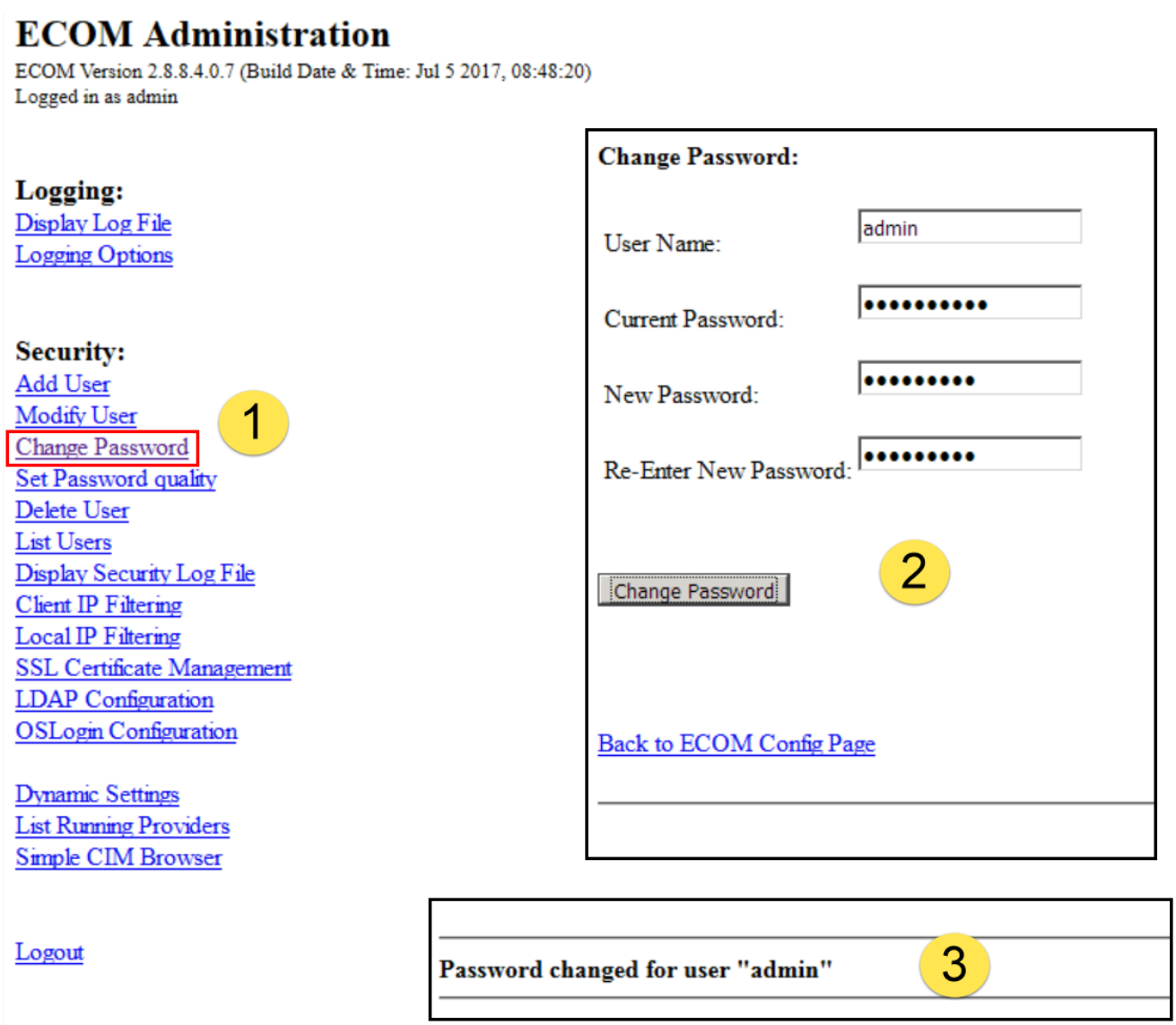

Figure 26. Changing the default admin password

In addition to changing the admin password, Dell EMC recommends creating a new user dedicated for VASA authentication from vCenter. Administrative access is required for VASA Provider registration with vCenter for vVols. Figure 27 shows the creation of a user account named “vvoluser” with the role type of “administrator”.

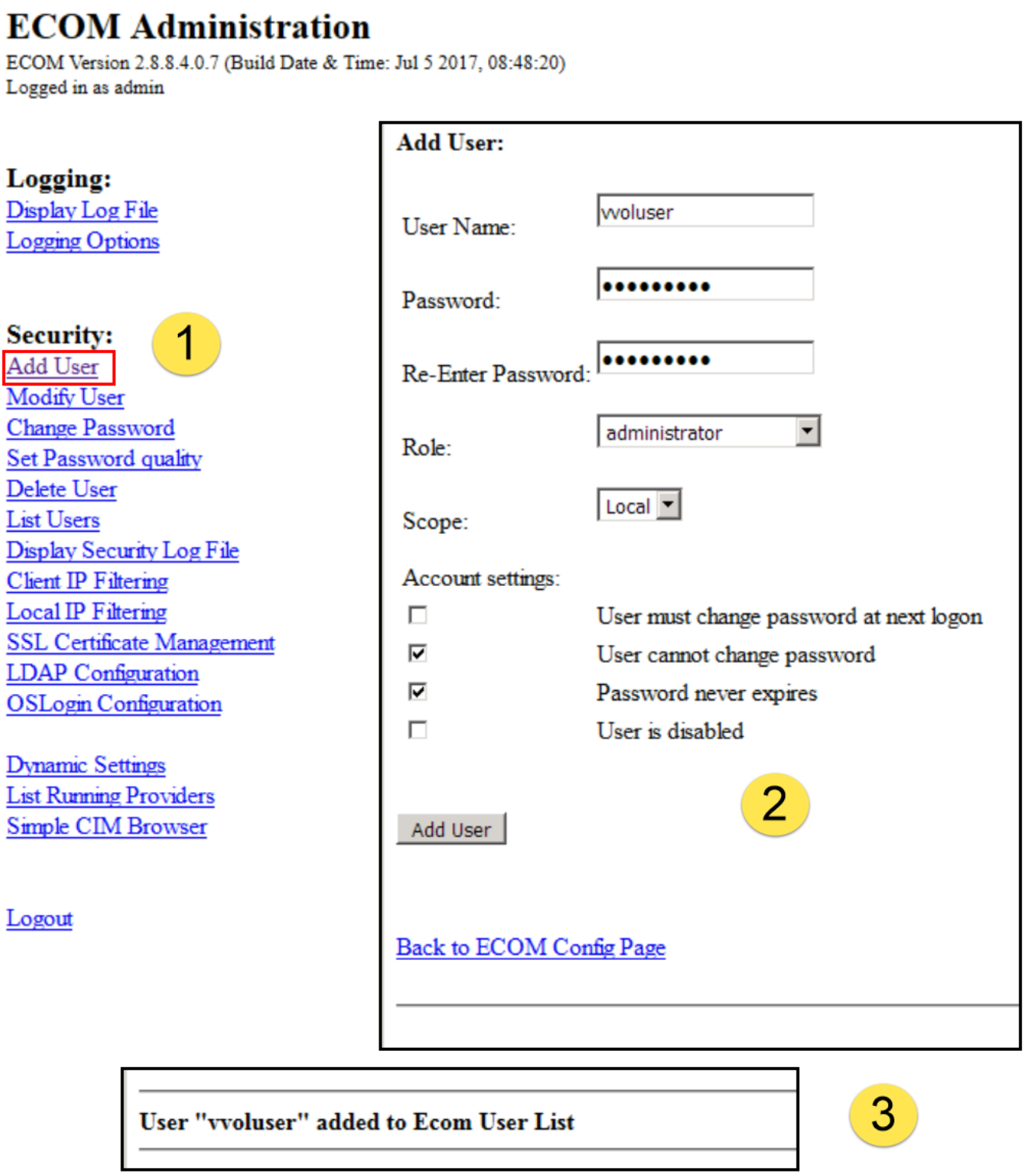

Figure 27. Creating a user for the VASA 2.0 Provider registration

Configuring Virtual Volumes on VMAX and PowerMax

The following sections will detail how the SAs can manage their vVol tasks through the GUI and the CLI.

Dell EMC does not support Virtual Volumes on external storage attached to a VMAX (e.g. Dell EMC CloudArray®).

Using Unisphere with Virtual Volumes

vVols has been integrated into Unisphere for VMAX and Unisphere for PowerMax through a dashboard: VVol (vVol) Dashboard[6]. This dashboard is the central location for managing vVols in a VMAX environment. From here, the storage administrator can create storage containers with the required storage resources, provision Protocol Endpoints to the ESXi hosts, and enter VASA Provider details to retrieve a status. The vVol dashboard for Unisphere for PowerMax 9.1 appears in Figure 28.

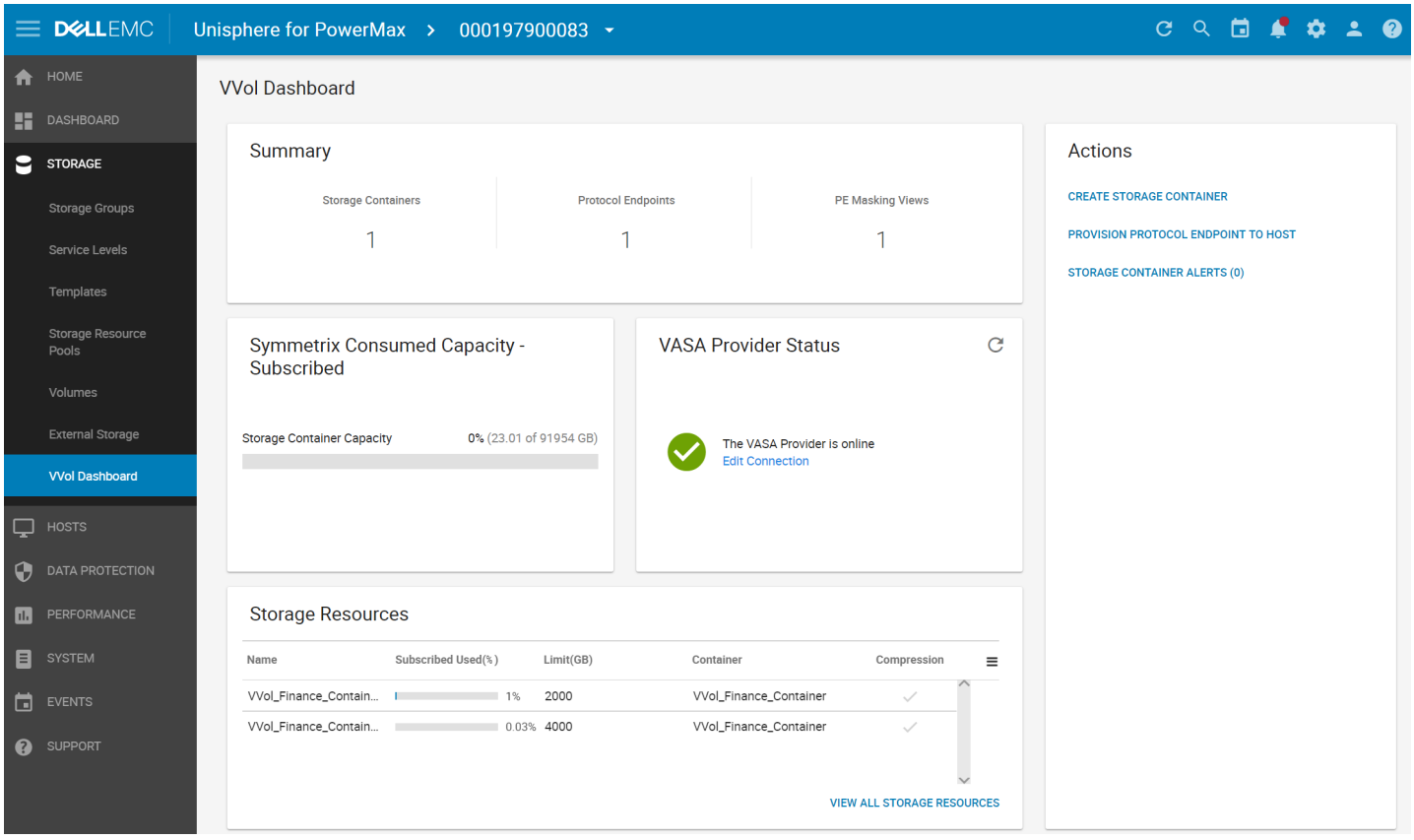

Figure 28. Unisphere for PowerMax 9.1 VVol dashboard

Adding the VASA Provider in Unisphere

The vVol dashboard provides a location to add the IP from the VASA Provider so that the SA can monitor the VP status seen in Figure 29. The status is obtained not simply

[6] The naming convention "VVol" was fixed to "vVol" in later versions of Unisphere for PowerMax.

by pinging the IP, but through a special call mechanism to the array which guarantees accurate results concerning VP viability. Should it report a problem, the SA can inform the VMware admin who can investigate on the VMware side. Note the VP Status is not a required component of vVols and has no bearing on its function.

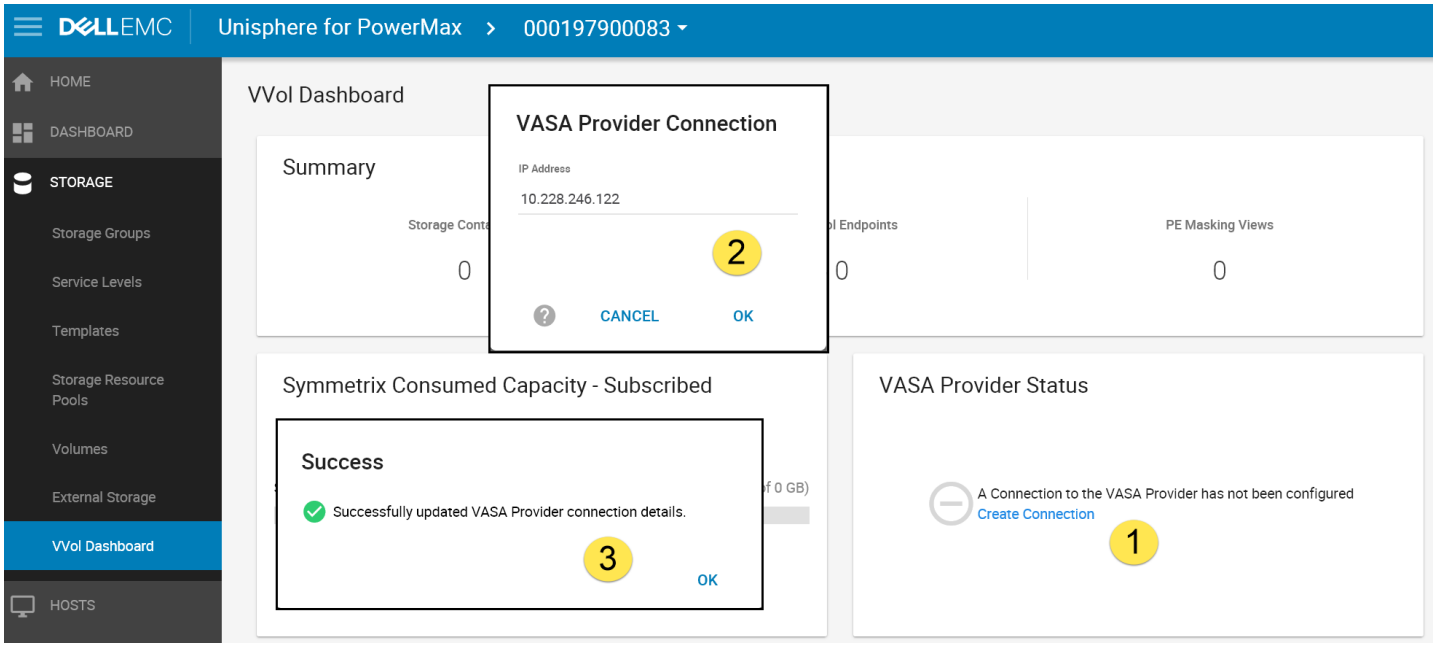

Figure 29. Adding VASA VP to Unisphere

Creating the Storage Container in Unisphere

As previously explained, a storage container (SC) is a logical construct on the array that partitions space based on SLs. The SA creates storage resources which at a high level are a combination of a service level and a storage size. A storage container may only have a single storage resource for each SL and workload type (if applicable) combination. An SC, for instance, may not have two storage resources with an SL of Optimized. Depending on the array model, a storage resource may also have an attribute of compression (plus deduplication on PowerMax) applied to it (default behavior).

Dell EMC supports 16 storage containers on a VMAX or PowerMax array. Multiple containers may be desired, for example, to separate test and dev environments from production, or limit storage for a particular business unit. Multiple containers do not change performance, however. Only the service level of a storage resource impacts performance.

The following walks through the SC wizard in individual steps.

SC Step 1

From the VVol dashboard, Figure 30, access the CREATE STORAGE CONTAINER option in the Actions menu on the right-hand side.

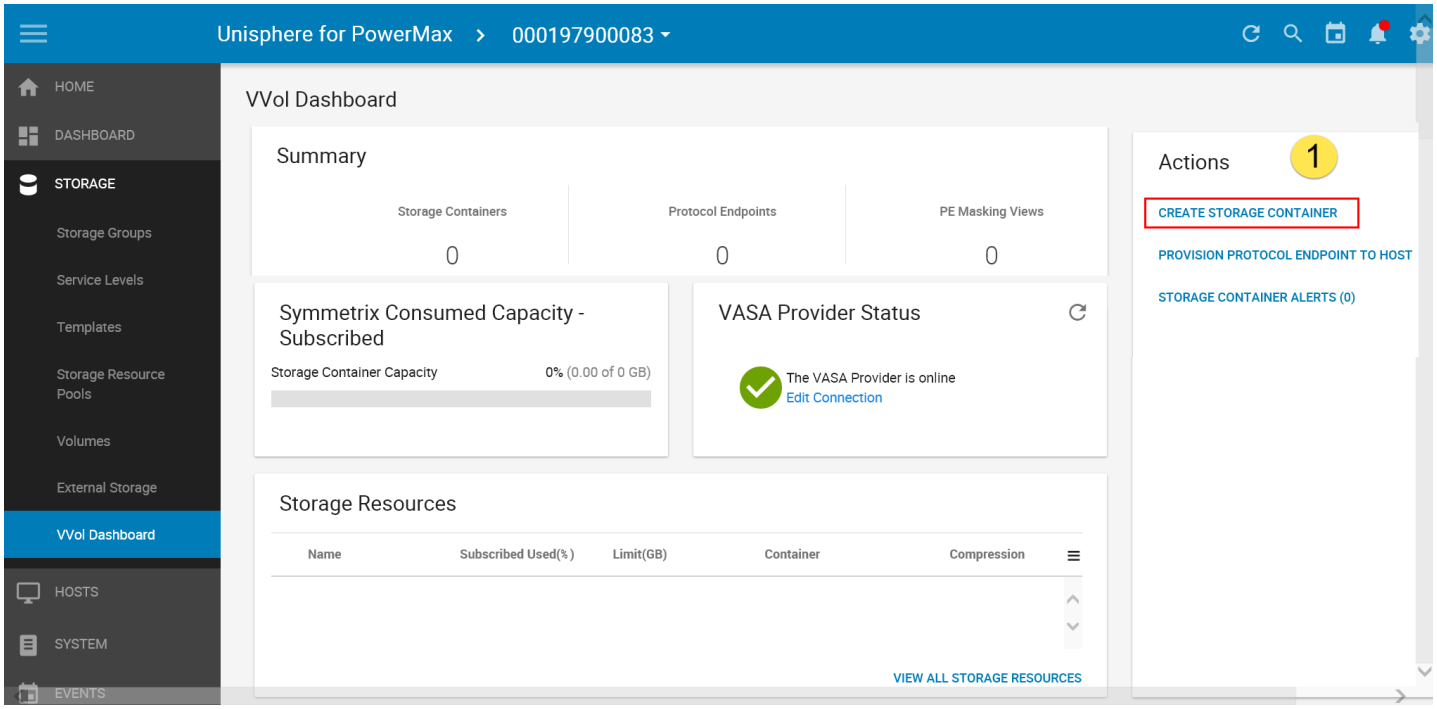

Figure 30. Create Storage Container – step 1

SC Step 2

Enter a name for the SC and, if desired, a description as in Figure 31. The description is only available to the SA. The VMware admin will not see it so it should not be used in the hopes it can provide important information within vSphere. Select Next.

Figure 31. Create Storage Container – step 2

SC Step 3

Add as many Storage Resources as desired for the SC Figure 32. Compression/deduplication is on by default, though the box can be unchecked if desired. Each time a line is complete, use the plus symbol on the right of the resource to add another line. The plus symbol will only appear when the cursor hovers near the end of the line. Unisphere will automatically generate a new name for each resource based on the SC name. If desired, change the name. All fields are required. An SC may have storage resources from multiple SRPs. The Limit represents the total storage in GB available for that SL. The wizard will prevent the user from having two storage resources with the same SLO/workload combination. As the PowerMax does not have the concept of a workload, only one SL per container is possible. If more than one storage resource with the same SL is desired, create a second storage container. Note that the compression attribute does not change the limitation of one SL type per storage container. For example, if a diamond SL resource with and without compression is desired, two storage containers are needed.

The Total Resource Subscribed Limit is a logical limit. Storage resources do not reserve space in the SRP. Consumers using the SRP outside of the vVol paradigm are not prevented from allocating beyond the size of free space available in the SC. However, vVol storage groups are included in the Storage Group Demand Report so actual usage is always reflected properly.

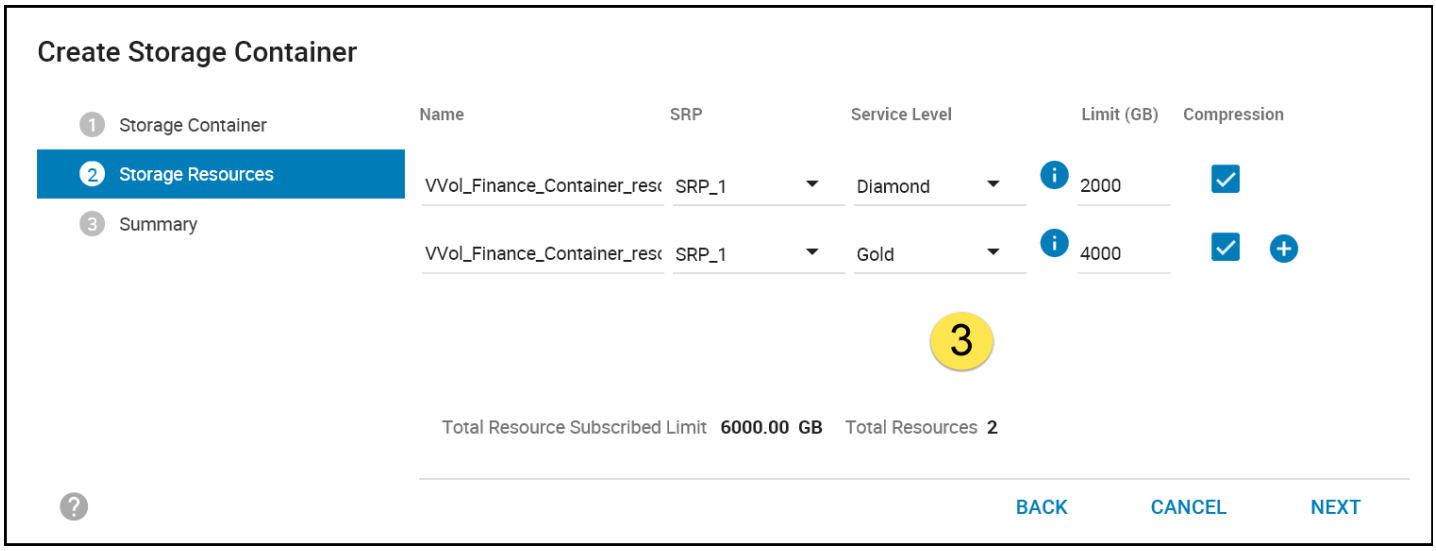

Figure 32. Create Storage Container – step 3

SC Step 4

Review the final screen in Figure 33 and select Run Now to create the SC.

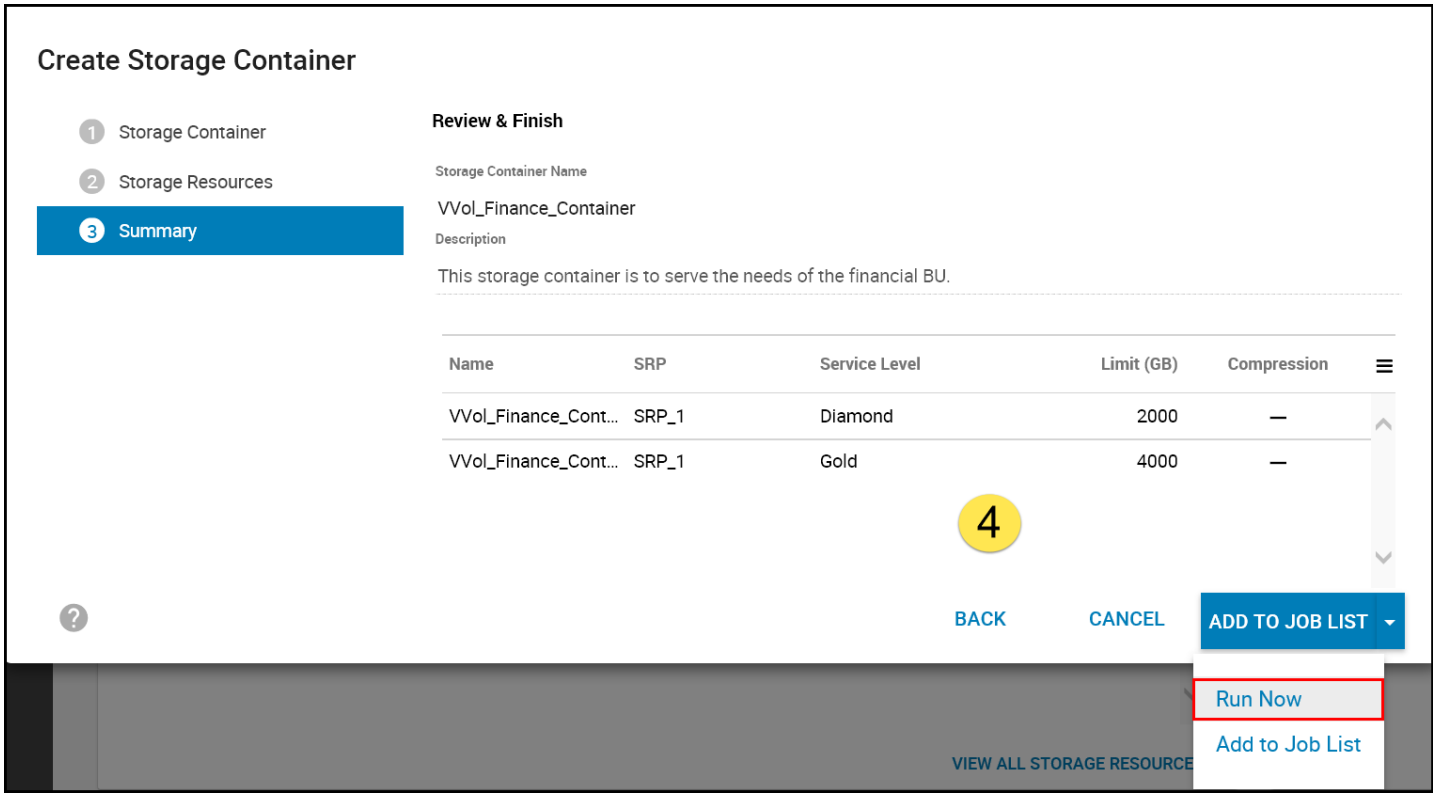

Figure 33. Create Storage Container – step 4

Each time a task is run in Unisphere, a dialog box (Figure 34) appears providing the details of the task.

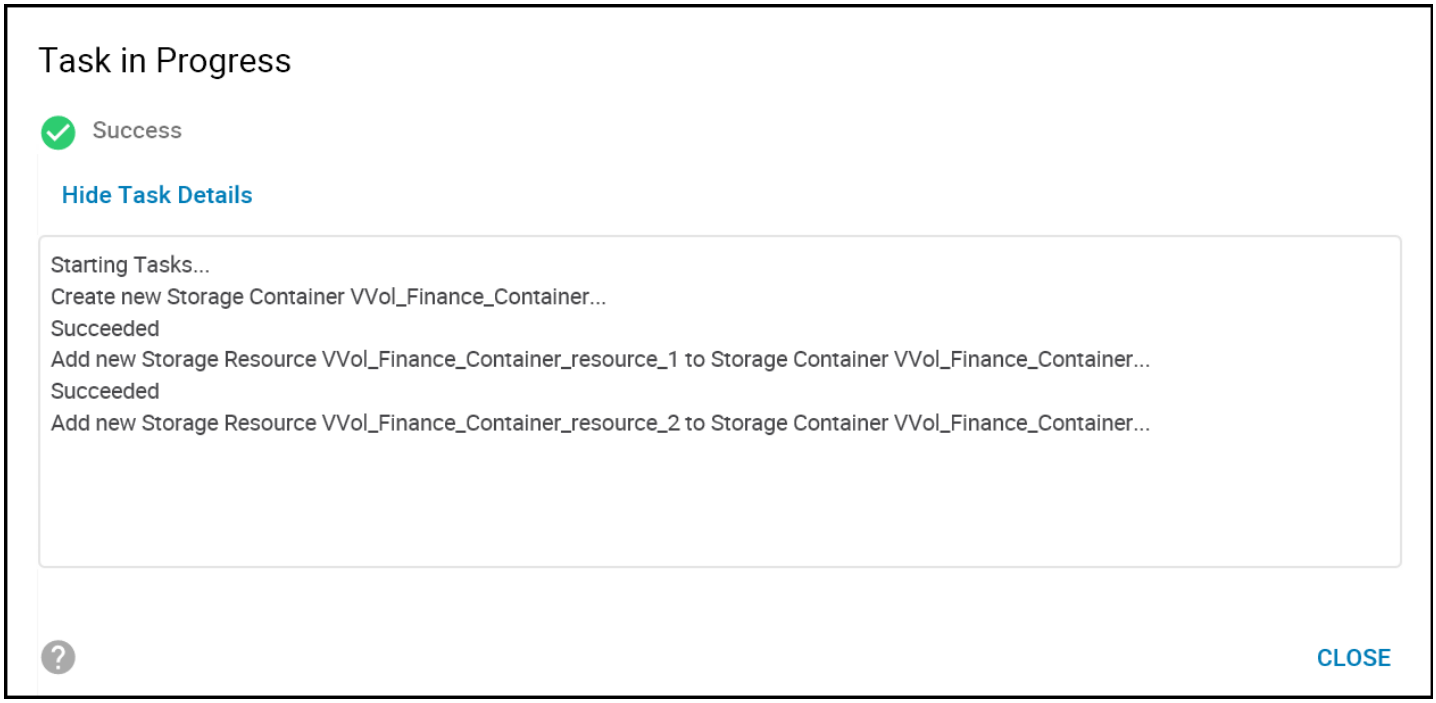

Figure 34. Create Storage Container – completion

Creating the Protocol Endpoint in Unisphere

The Protocol Endpoint is a small device that is used to enable IO between vSphere and the vVols on the array. vVols are bound and unbound to the PE by the VASA Provider, but once a vVol is bound to the PE, the VP is not required for IO to take place between the VM and the array.[7] This means even if the VP crashes, IO continues. Each ESXi host must be presented a unique PE to support vVols on the VMAX. Each ESXi host in a cluster may not see any PE but the one uniquely presented to it.

It should be noted that the PE, like vVols, uses a different World Wide Name (WWN) than a traditional device. The new format is known as a mobility safe ID (MID. Figure 35 shows a traditional WWN and Figure 36 shows a mobility safe ID.

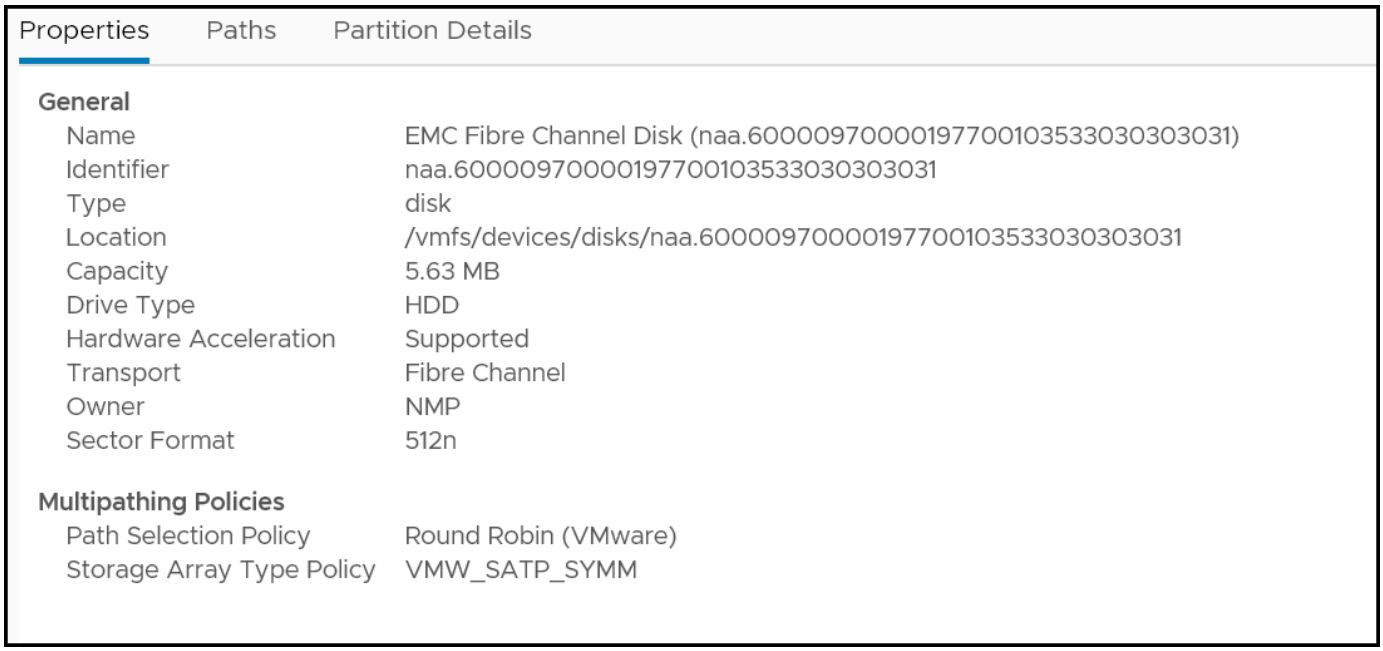

Figure 35. Traditional WWN format

[7] The VP is required to conduct any type of management activity such as adding new vVols, snapshots, powering on or off the VM, or deleting the VM, among others.

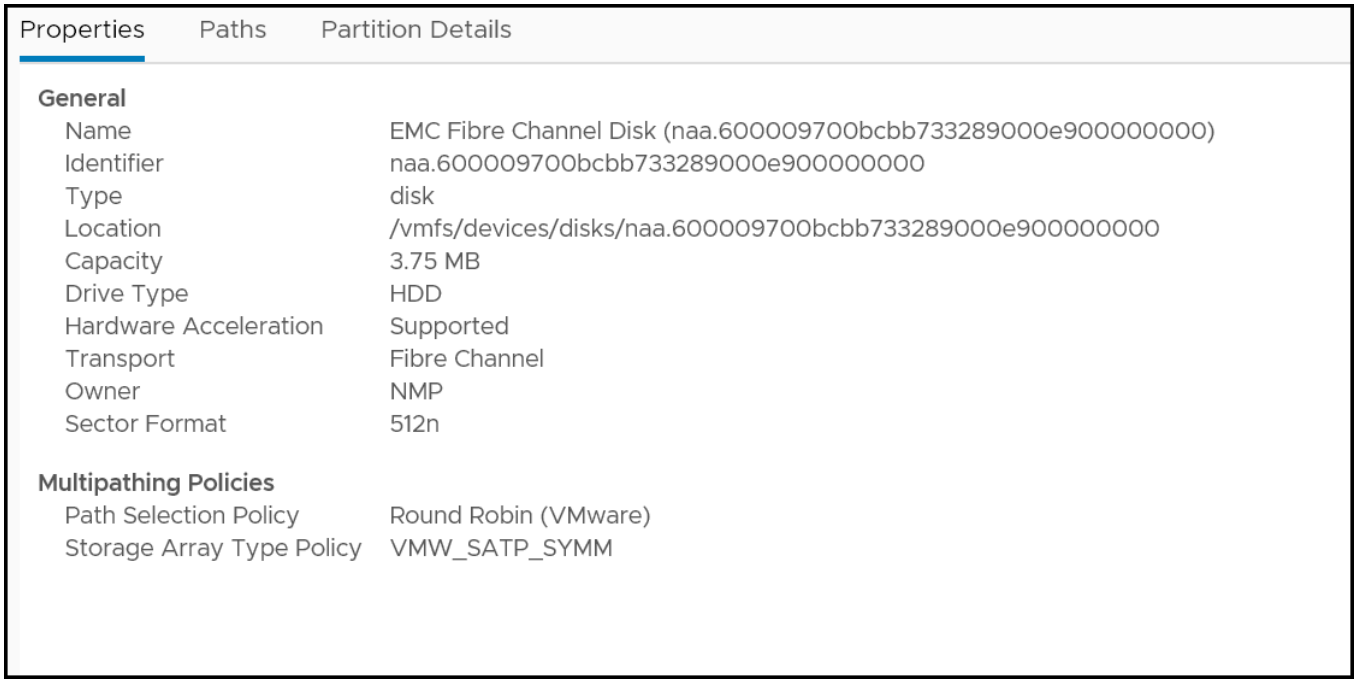

Figure 36. Mobility safe ID WWN format

PE devices with mobility safe ID will not advertise ALUA support. This ensures the pathing software will not recognize the PE as an ALUA device.

The VASA 2.0 Provider does not support iSCSI for the PE. Only FC is supported. iSCSI is supported with the VASA 3.0 (EVASA) Provider.

PE Step 1

From the VVol dashboard, Figure 37, navigate to PROVISION PROTOCOL ENDPOINT TO HOST in the Actions section.

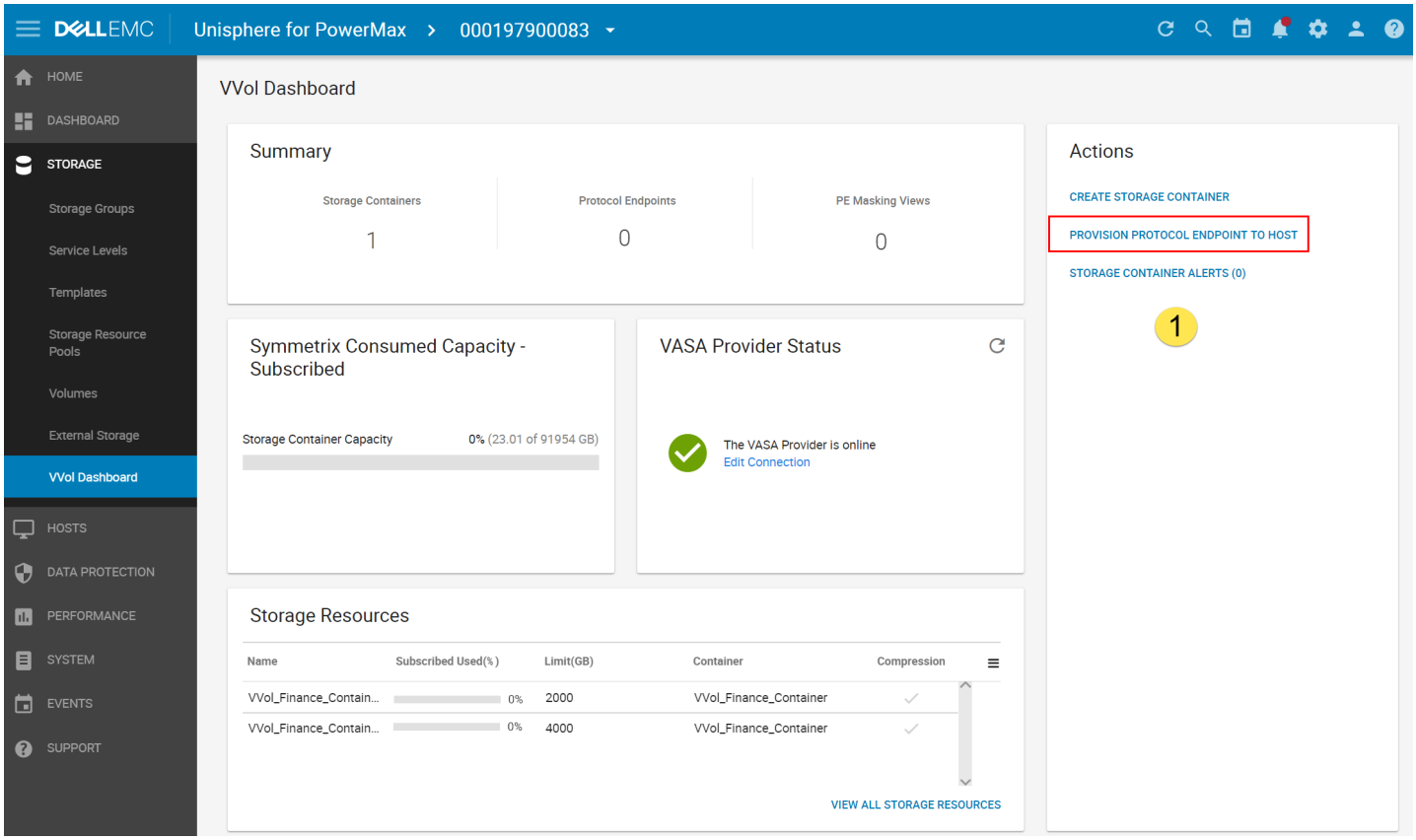

Figure 37. Provision Protocol Endpoint – step 1

PE Step 2

Start by selecting the correct initiator group (host group) for the ESXi host to which the PE will be provisioned. Be sure the group is comprised of only FC initiators from the host in question. Initiators from multiple hosts in a single group or cascaded initiator groups are not supported. The step is shown in Figure 38.

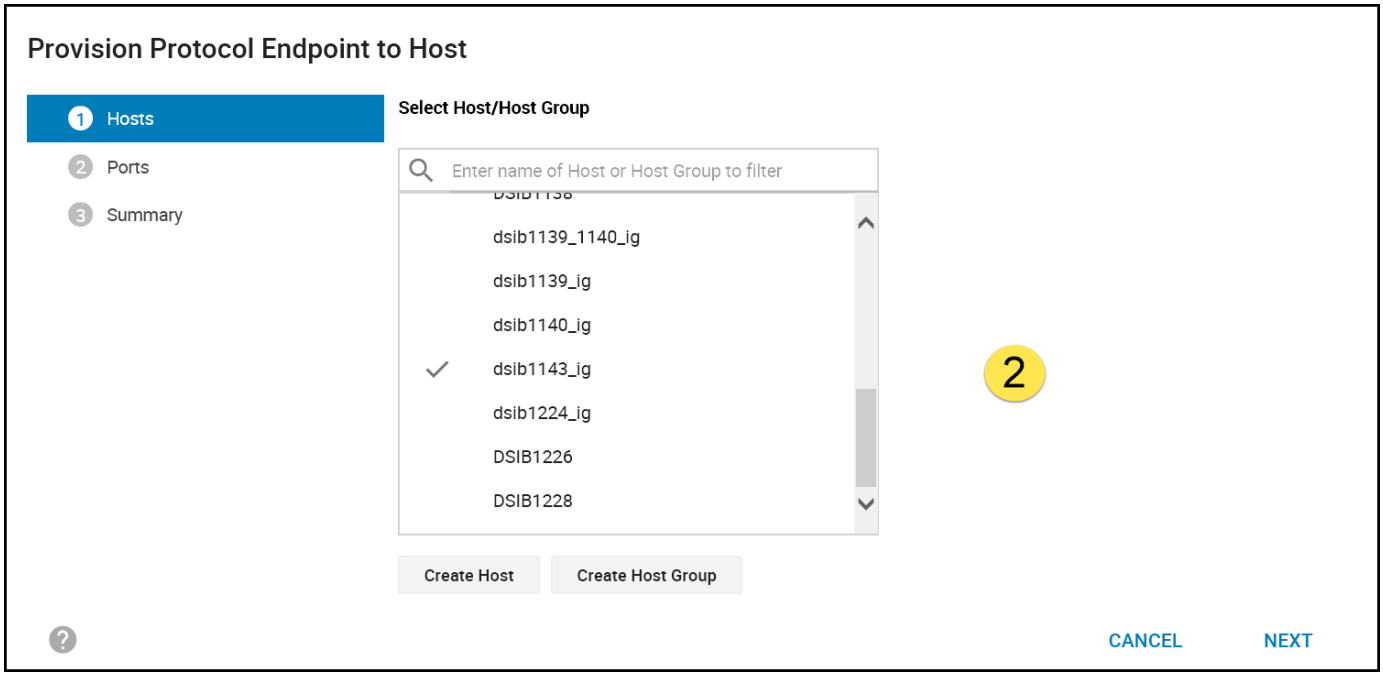

Figure 38. Provision Protocol Endpoint – step 2

Beginning with Unisphere for PowerMax in Figure 39, the user is warned that each ESXi host may only have one PE and it cannot be shared.

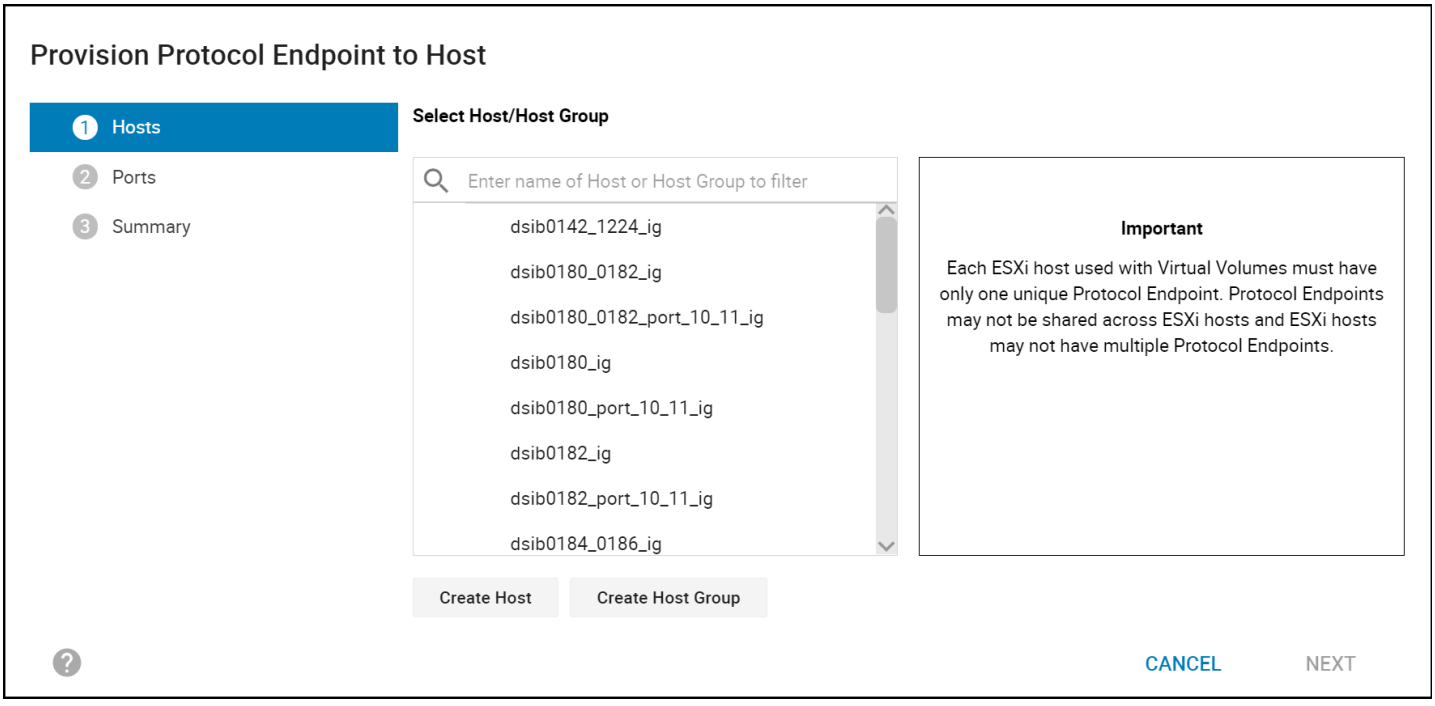

Figure 39. Provision PE in Unisphere for PowerMax 9.2

PE Step 3

In step 3, Figure 40, select the port group which contains the zoned ports. The wizard offers the user the chance to create a new port group if desired. If selected, Unisphere

will show the ports which are currently zoned and allow the user to choose which ones to add to the group.

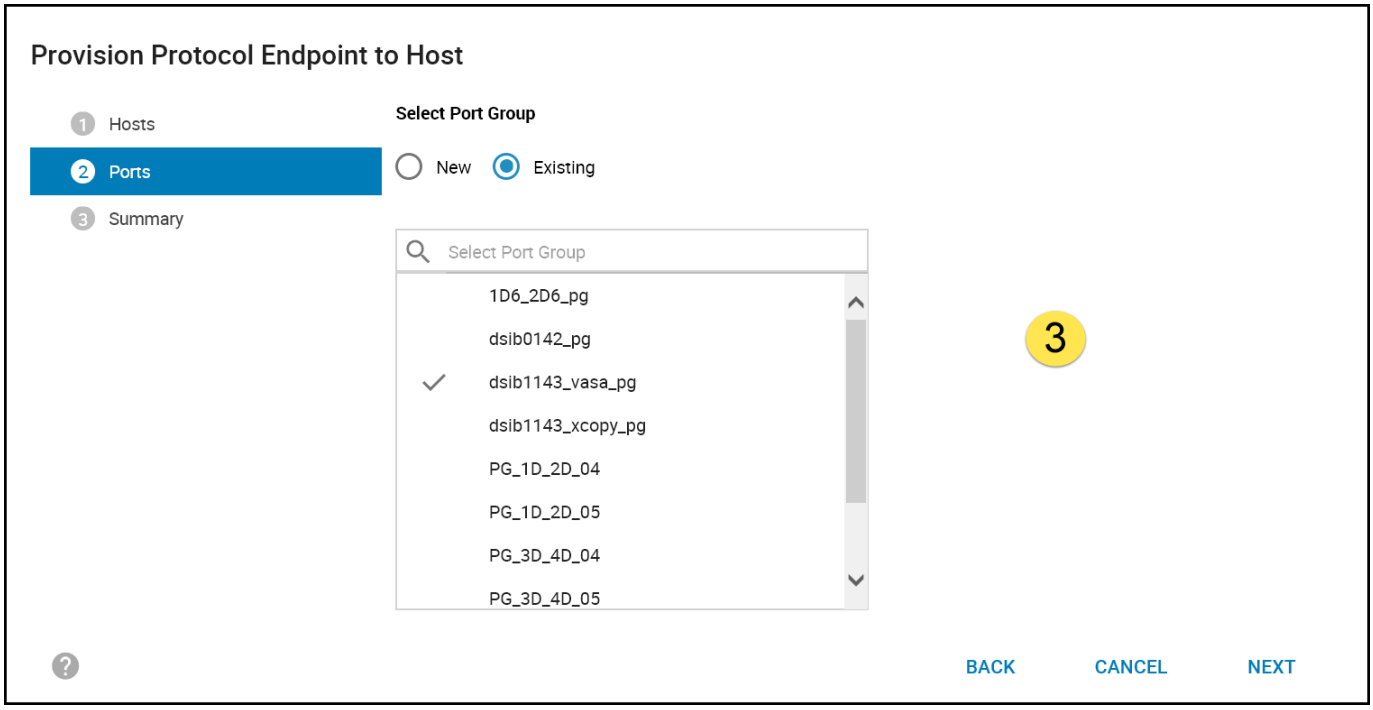

Figure 40. Provision Protocol Endpoint – step 3

PE Step 4

In the final review, Unisphere will automatically generate a name for the masking view and storage group based on the initiator group name. In Figure 41 they have been renamed to better reflect the purpose of the masking view. When the job is run, Unisphere will create a PE device and add it to the new storage group, create the port group if needed, and finally create the masking view.

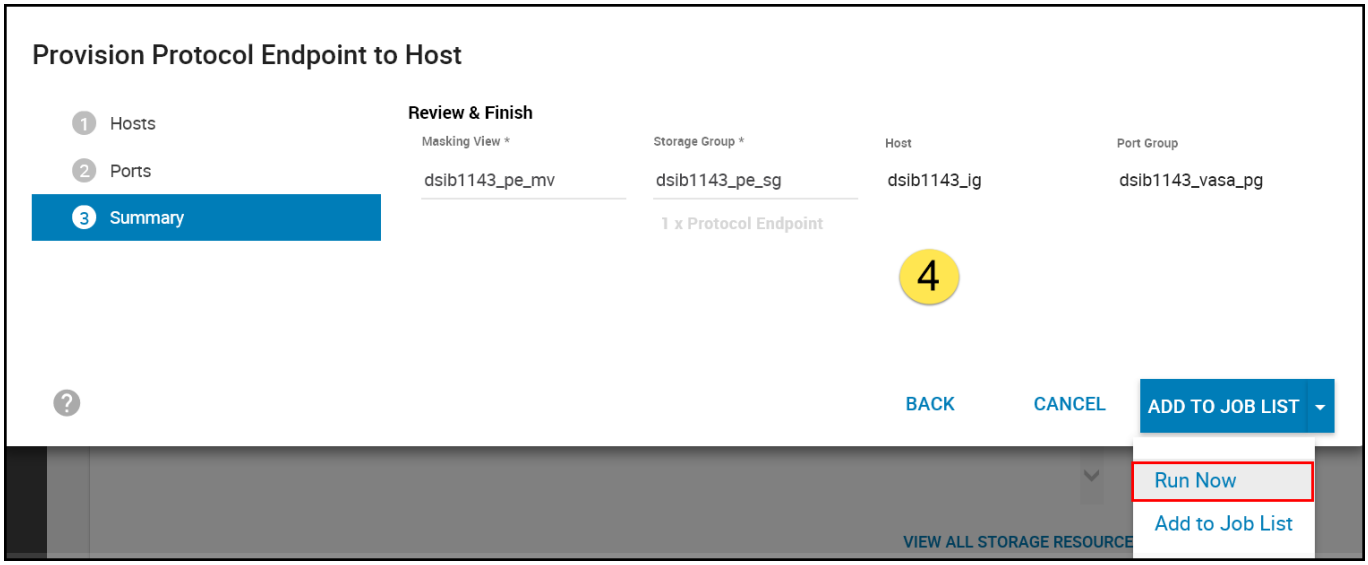

Figure 41. Provision Protocol Endpoint – step 4

In the vSphere Client the PE device appears as a 3.75 MB TDEV shown in Figure 42. Note the mobility safe ID.

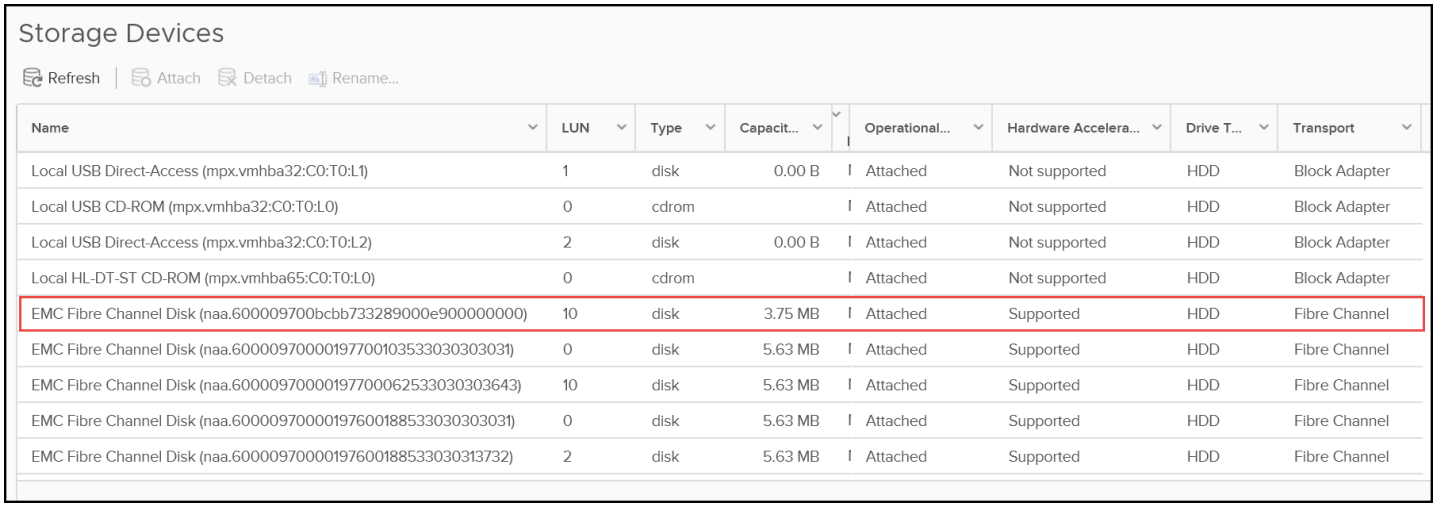

Figure 42. PE device in vSphere

There is also a specific screen for Protocol Endpoints which can be accessed in Figure 43.

Note that the PE will not appear in its designated location (Figure 43) in the vSphere Client until the vVol datastore is created.

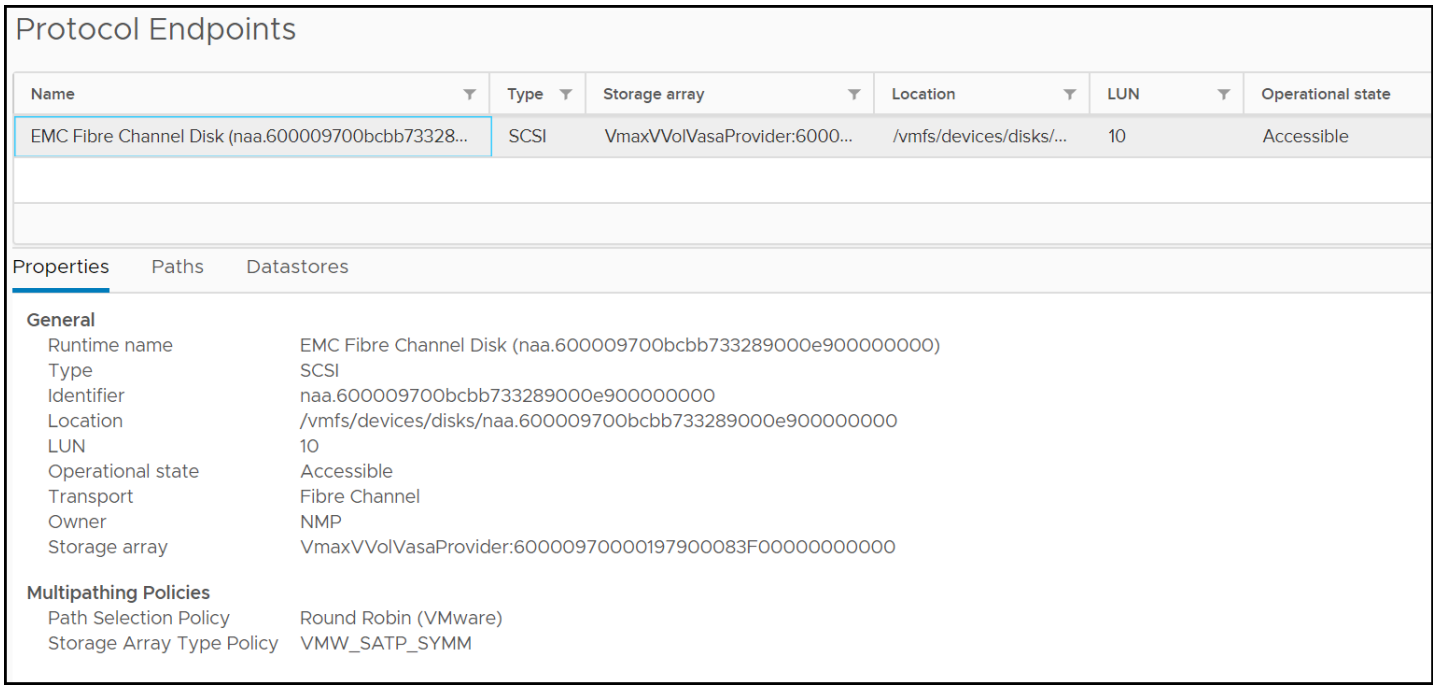

Figure 43. Protocol Endpoints sub-tab

Using Solutions Enabler with Virtual Volumes

In addition to Unisphere, vVols may be managed through Solutions Enabler. Almost all capabilities that exist in Unisphere for vVols are available through the CLI, though not in “wizard” format. The exception is the VASA Provider Status which has no associated command. The following will detail the commands available to the user through SE for vVols. For each creation statement a deletion statement follows in parentheses. The CLI examples are using a VMAX3, therefore the workload type is specified. For VMAX All Flash and PowerMax arrays, no workload should be specified.

Creation of the Storage Container

There are two parts to creating the storage container: the container object and the storage resource object. A storage container by itself has no actual storage associated with it, rather it is a logical grouping of storage resources. Storage resource objects, represented by an SLO and size, are added to the container to provide the storage from which to provision vVols.

Storage Container

To create the container:

```
symcfg create -sc -name Demo_VVol_Container -type vvols -
description "This is a demo vVol Container" -sid 56

(symcfg -sid 56 delete -sc -sc_name Demo_VVol_Container)
```

There is no response to the command. To list the container:

```
symcfg list -sc -v -sid 56
```

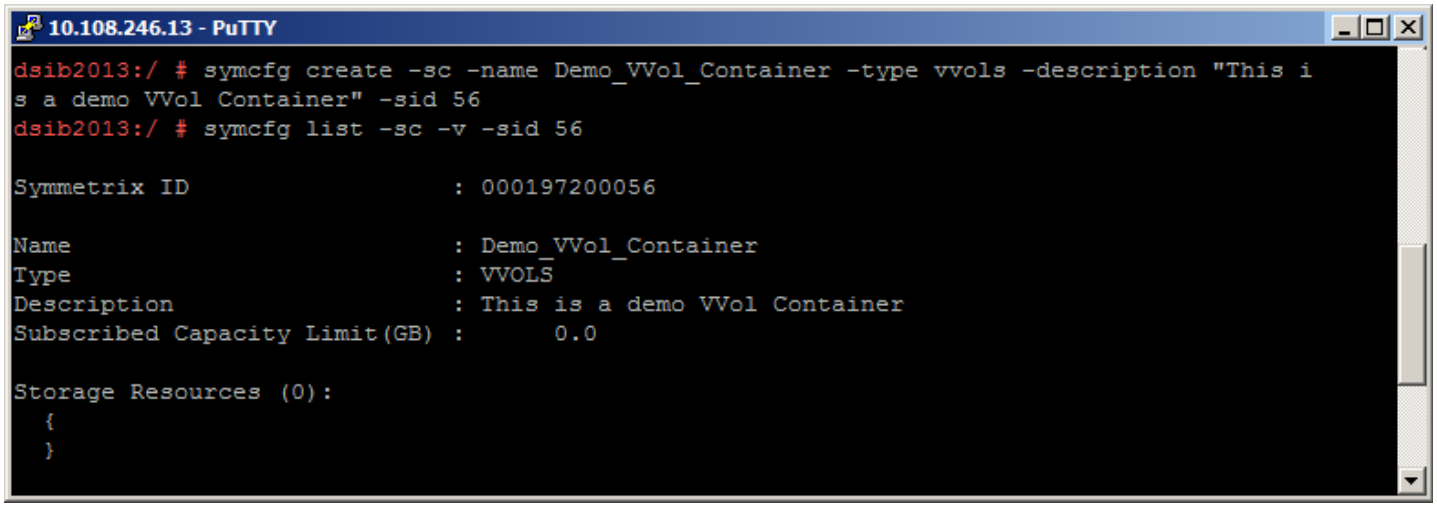

Figure 44. Create and list storage container

Storage Resources

Once the container is created, storage resources can be added. For each SRP and SLO combination, except Optimized (you cannot use “None” as the SLO), you can add 3 storage resources – 2 with workloads (OLTP, DSS) and one without. The subscribed maximum defaults to GB.

```
symcfg -sid 56 -sc -sc_name Demo_VVol_Container add -sresource
Gold_Resource -slo Gold -wl OLTP -subscribed_max 1024
```

```
(symcfg -sc -sc_name Demo_VVol_Container remove -sresource
Gold_Resource -sid 56)
```

```
10.108.246.13 - PuTTY
dsib2013:/ # symcfg -sc -sc_name Demo_VVol_Container add -sresource Gold3_Resource -slo Gold  -subs
cribed_max 1024 -sid 56
dsib2013:/ # symcfg list -sc -v -sid 56

Symmetrix ID                  : 000197200056

Name                          : Demo_VVol_Container
Type                          : VVOLS
Description                   : This is a demo VVol Container
Subscribed Capacity Limit(GB) :   1024.0

Storage Resources (1):
  {
   ------------------------------------------------------------------------
                                                                  Capacity
                                                                 Limit Subs
   Name                     SLO Name        Workload SRP Name         (GB)
   ------------------------ --------------- -------- --------------- ----------
   Gold_Resource            Gold            OLTP     SRP_1             1024.0
                                                                     ----------
   Total                                                             1024.0
  }
```

Figure 45. Add storage resources to container

If the –detail flag is added to the list command, the subscribed usage is displayed as in Figure 46.

```
10.108.246.13 - PuTTY
dsib2013:/ # symcfg list -sc -v -sid 56 -detail

Symmetrix ID                  : 000197200056

Name                          : Demo_VVol_Container
Type                          : VVOLS
Description                   : This is a demo VVol Container
Subscribed Capacity Limit(GB) :   1524.0
Subscribed Capacity (GB)      :      0.0
Subscribed Capacity (%)       :      0


Storage Resources (2):
  {
   ------------------------------------------------------------------------------------------
                                                                  Capacity
                                                                 Limit Subs          Subs
   Name                     SLO Name        Workload SRP Name         (GB)        (GB) (%)
   ------------------------ --------------- -------- --------------- ---------- ----------
   Diamond_Resource         Diamond         OLTP     SRP_1              500.0        0.0   0
   Gold_Resource            Gold            DSS      SRP_1             1024.0        0.0   0
                                                                     ---------- ----------
   Total                                                             1524.0        0.0   0
  }
```

Figure 46. Listing storage usage of container on VMAX

If compression is available on the array, a separate column will be shown as in Figure 47.

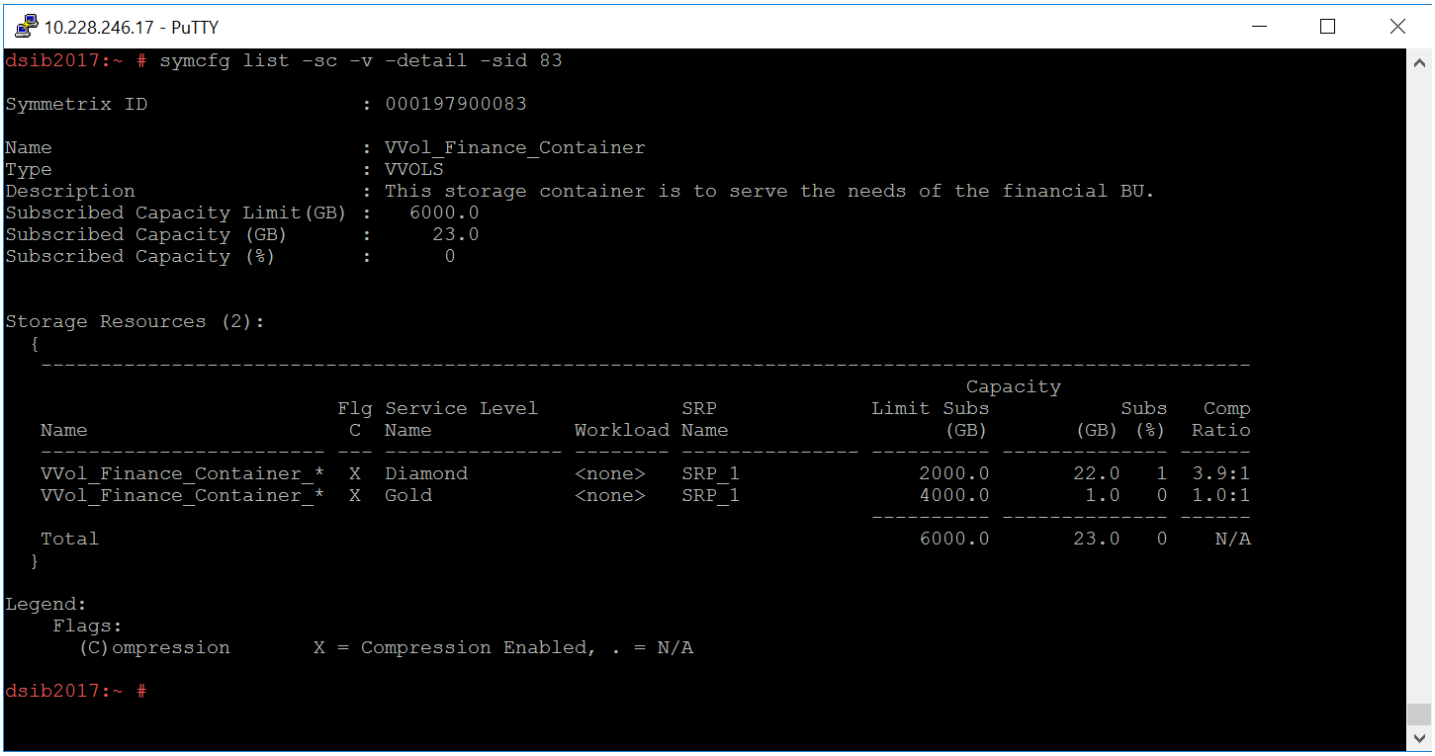

Figure 47. Listing storage usage of container on PowerMax

Creation of the PE

To create a PE device:

```
symdev create -pedev -sid xxx -v
```

```
(symdev delete 58 -sid 56)
```

Note the –v (verbose) flag is not required, however without it the device ID will not be returned. Figure 48 shows the output.

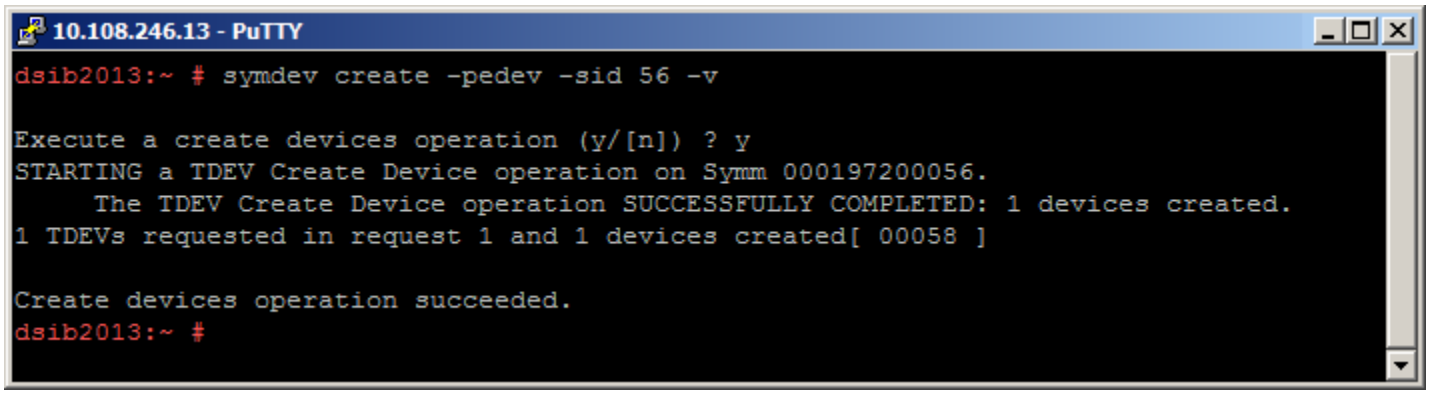

Figure 48. PE device creation

A PE device is like any other TDEV in that once created it can be added to a storage group and then presented to an ESXi host. Remember that each ESXi host must see its own unique PE device and the initiator group for that masking view may only contain FC initiators of that host (no cascading or initiators from other ESXi hosts).

Host IO Limits/Storage IO Control (SIOC)

Host IO Limits is a feature of the VMAX and PowerMax that allow users to place limits on the front-end bandwidth and the IOPS consumed by applications. Currently, vVols do not support the use of Host IO Limits. While it is possible to set a Host IO limit on the storage group that contains the Protocol Endpoint, it will have no bearing on the vVol IOs.

vVols support the use of VMware Storage I/O Control at the Storage Policy level, providing a way to limit IO to virtual machines. Figure 49 demonstrates the functionality.

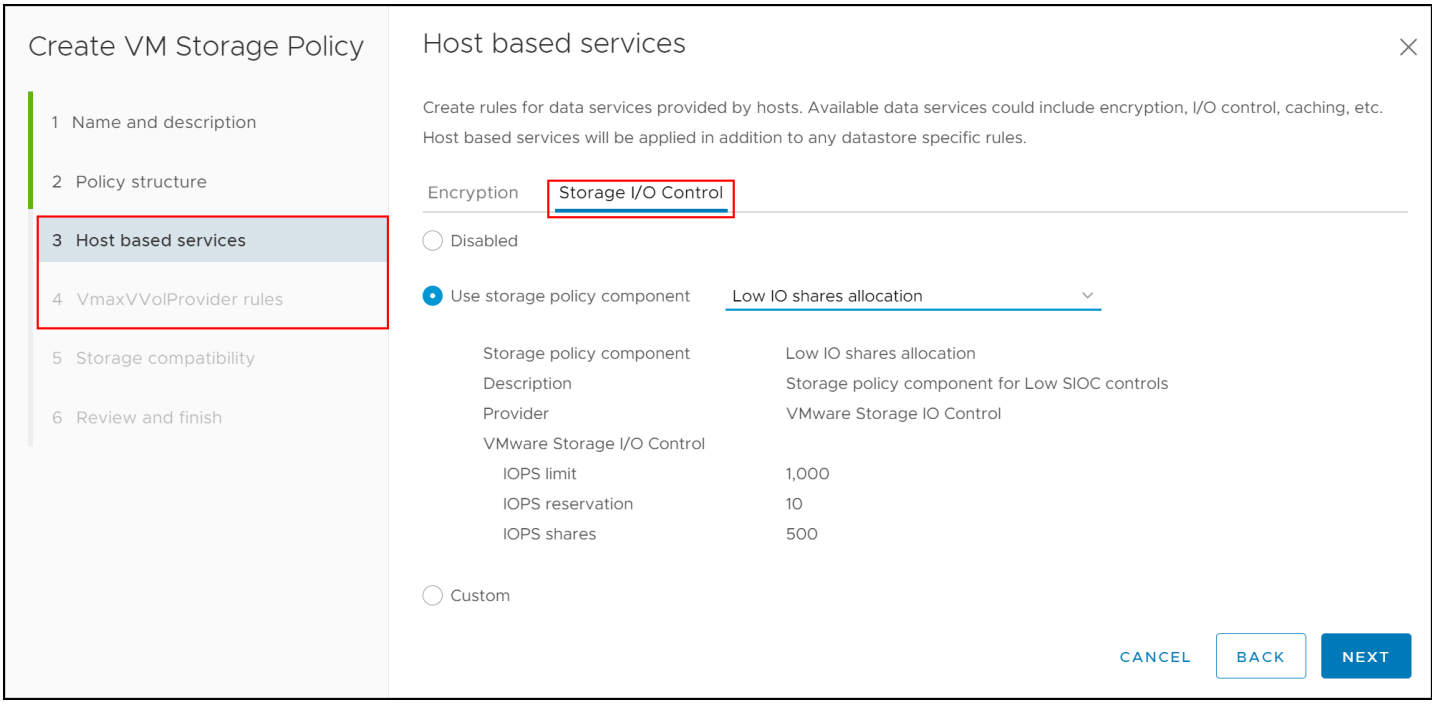

Figure 49. Storage I/O Control with vVols

Registering the VASA Provider in vCenter

The VASA Provider (VP) is the orchestration entity behind vVols. It enables most functions related to vVols including creation, deletion, powering on/off, etc. The primary function of a VM, IO, however, does not require the VP once the vVol is bound and the VM running. Once the VP is deployed, it must be registered in the vSphere vCenter so VMware can communicate with it.

VP Registration Step 1

Start by accessing the Storage icon in the Home page of the vSphere Client in Figure 50.[8]

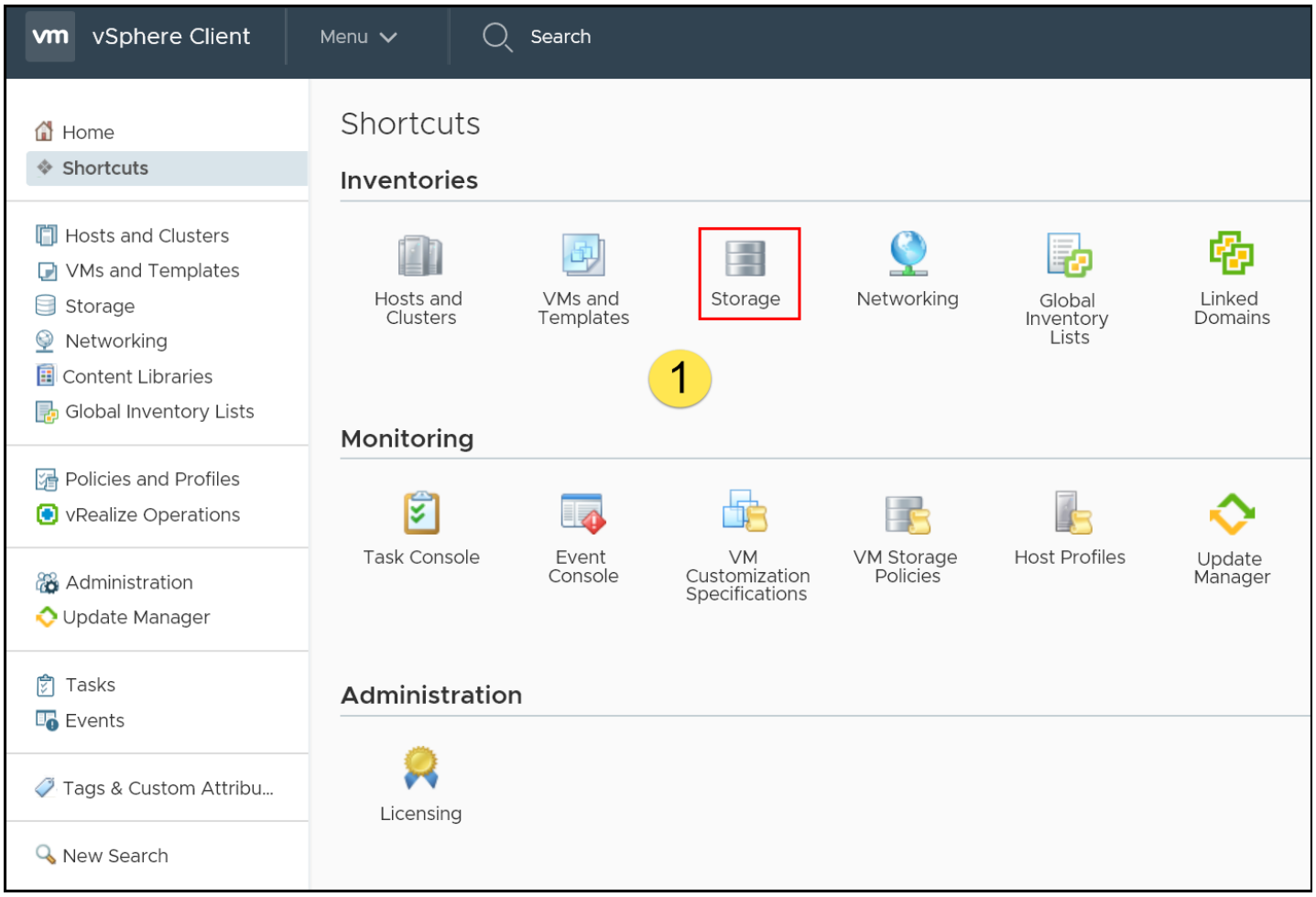

Figure 50. Registering the VASA Provider - step 1

VP Registration Step 2

Now at the vCenter level, select the Configure tab on the right, and then Storage Providers on the left-hand side menu shown in Figure 51. Now select + Add to open the dialog for the VASA Provider.

[8] Although the VASA Provider can be registered in the vSphere thick client in 6.0, EMC recommends using the vSphere Web Client or vSphere Client (HTML5) for all activities related to vVols.

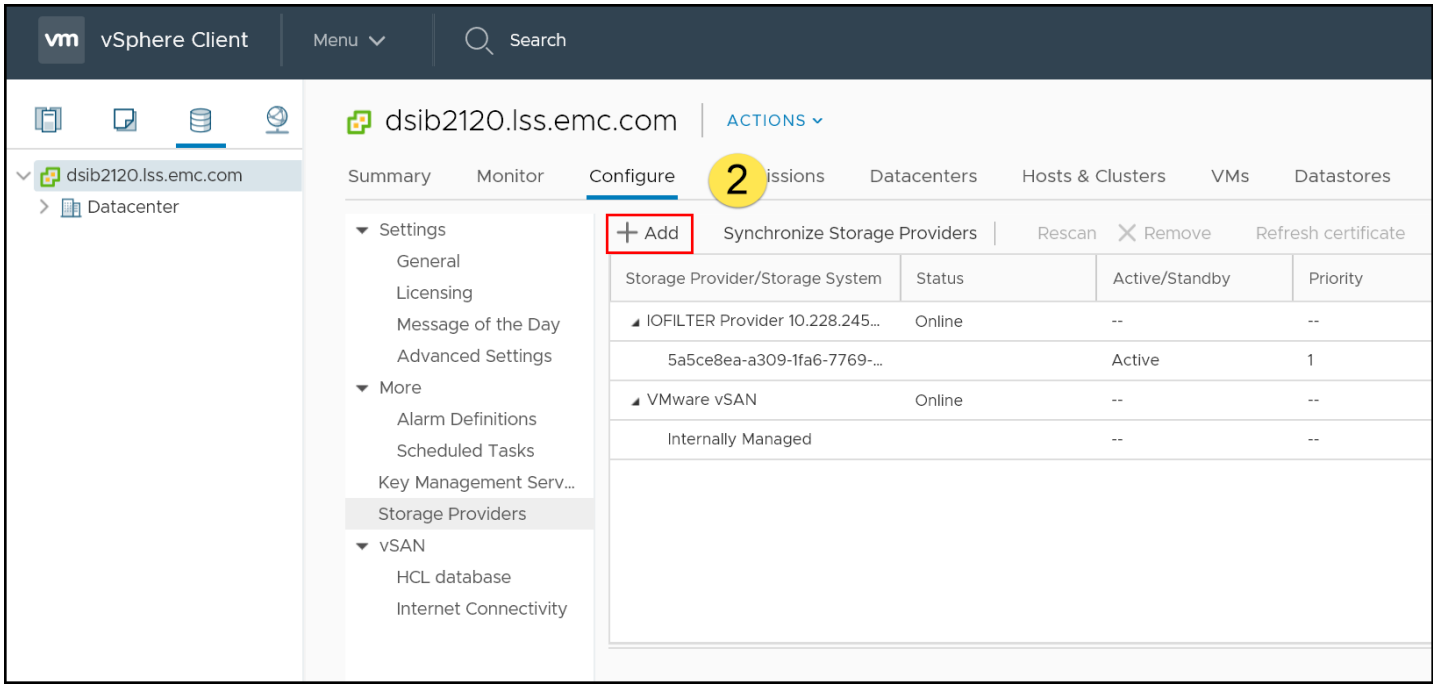

Figure 51. Registering the VASA Provider - step 2

VP Registration Step 3

In Figure 52 enter the VP information in the dialog box. The dialog box has 4 fields:

- Name: Any descriptive name
- URL: This is the VP URL for the deployed appliance. The URL can be easily retrieved from the initial Appliance Info screen seen in Figure 11. In the Operations portion of the Dashboard the URL can be copied directly (step 3) and pasted into the URL field.
- User name: The user must be one with administrative privilege. In this example the newly created user, “vvoluser”, is used.
- Password: The user password.

An additional checkbox is available to use a storage provider certificate. Dell EMC does not currently support using the non-default certificate. Once the fields are filled, select OK.

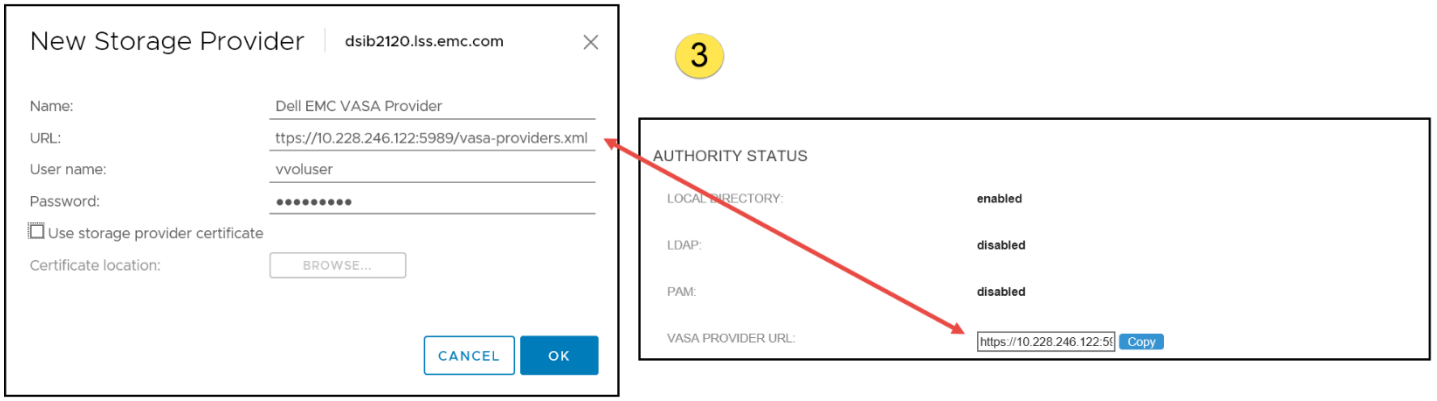

Figure 52. Registering the VASA Provider – step 3

VP Registration Step 4

In step 4, Figure 53, VMware returns a certificate error indicating the VP host is not trusted. Select Yes to accept the certificate. This error is expected. This error can be avoided by importing the vCenter certificate into ECOM before registration, though it is unnecessary if you trust the host.

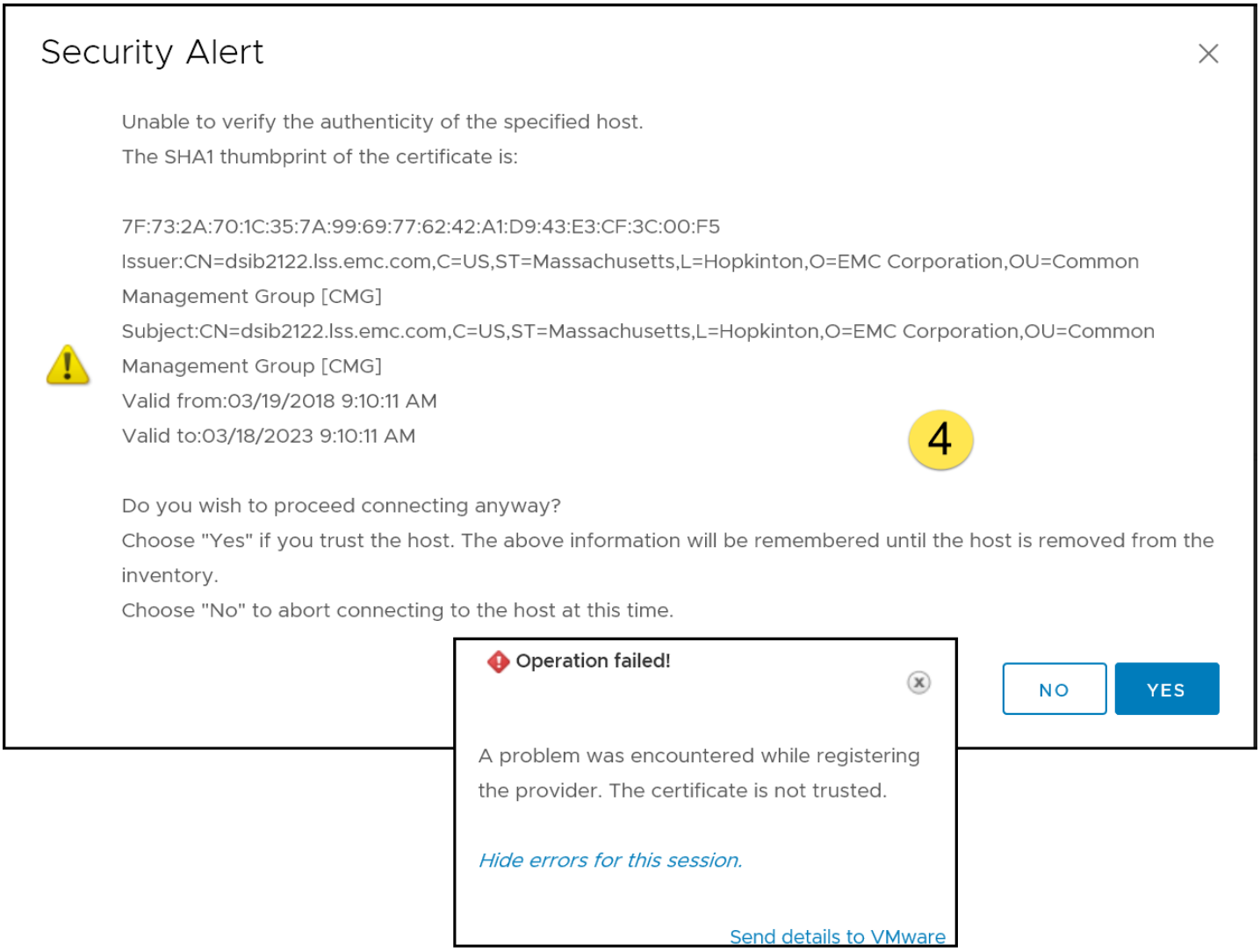

Figure 53. Registering the VASA Provider - step 4

VP Registration Completion

Once the registration succeeds, the VP will show that the array is online. It will appear similar to Figure 54. If it does not show as online, a Rescan can be run, though it should be unnecessary.

Registration of the VASA Provider is not blocked if the Protocol Endpoint is not presented to the host so a successful registration should not be used as an indication that the PE is available.

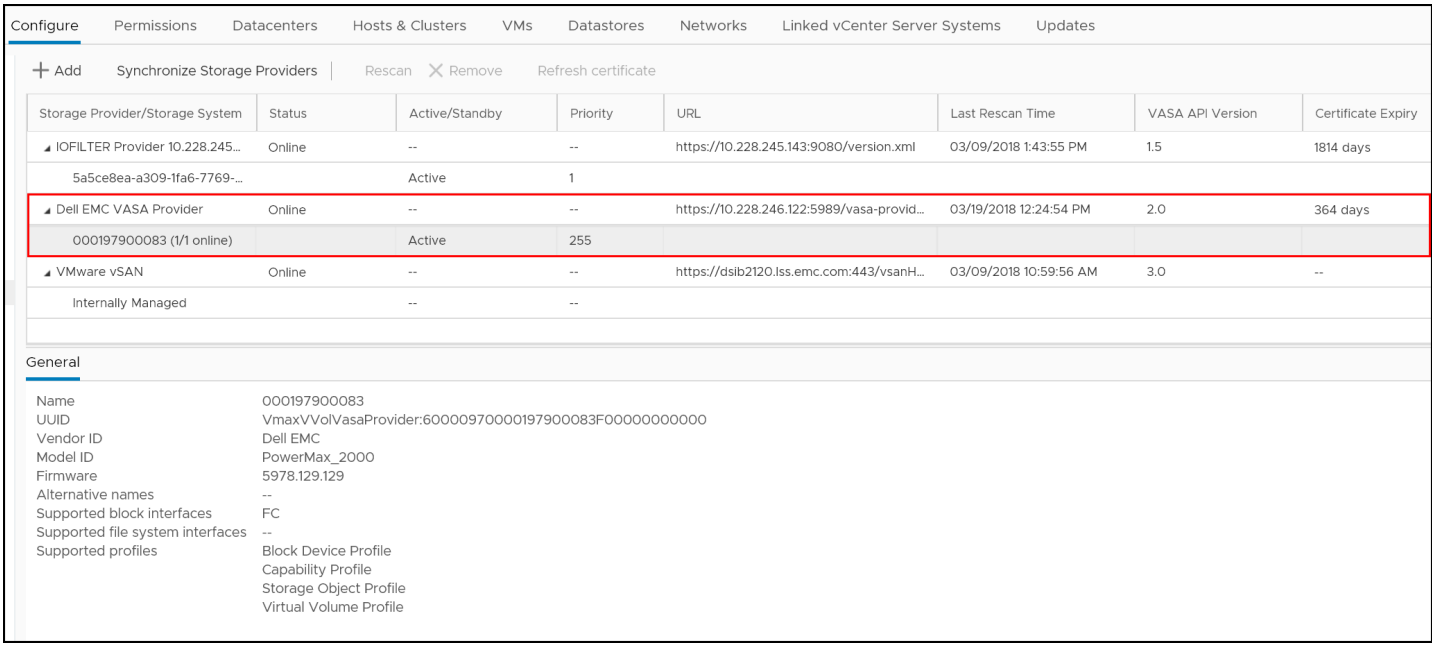

Figure 54. Registering the VASA Provider - completion

Multiple vCenters and certificates

If the VASA Provider will be registered in two vCenters (max supported)[9], a couple changes are required. These changes need to be made before registering the VPs in either vCenter, otherwise connectivity issues will ensue due to the certificate. The steps are outlined below.

RETAIN VP CERTIFICATE

The first step is to change a flag in the VASA Provider configuration seen in Figure 24. In this screen use the drop-down box next to "RETAIN VP CERTIFICATE" to change it from FALSE to TRUE and select "SET". The user will be alerted that the ECOM service must be restarted. Do this on the VASA Provider and then proceed to the next section.

Self-signed certificate

When the RETAIN VP CERTIFICATE parameter is set to TRUE, it is not possible to use the default self-signed ECOM certificate. This is because VMware requires that the key *CertificateSign* has a value in the certificate. The ECOM one does not. Therefore, a new certificate must be generated for each VASA Provider. Follow these steps to generate the new certificate on each VASA Provider:

1. Log in to the ECOM Config page (Figure 25) as admin.
2. Go to SSL Certificate Management and then select Option #2 Generate Self-Signed Certificate.
3. The Self-Signed Certificate page is displayed, enter the following details:

[9] If more than 2 vCenters are required, a request can be made by opening an SR with Dell EMC Support.

- Common Name: ‹hostname of VASA Provider›
- Country: ‹Country›
- State: ‹state›
- Locality: ‹locality›
- Organization name: ‹Org name›
- Organization Unit name: ‹Org unit name›
- Serial Number: 0
- SAN Email Address: ‹empty›
- SAN IP: ‹Vasa Provider IP address›
- SAN URI: ‹empty›
- Key Usage: DigitalSignature;NonRepudiation;KeyEncipherment;KeyAgreement;EncipherOnly;DecipherOnly;CertificateSign
- CA: ‹leave this unchecked›

4. Click Generate a Self-Signed Certificate.

5. Restart ECOM.

6. Register both VASA Provider in each vCenter. If already registered, unregister and register again.

Creating a vVol Datastore

Creating the vVol datastore is the final step in the vVol setup. It relies both on successful registration of the VASA Provider and presentation of the Protocol Endpoint to the host. If the VASA Provider is registered but the Protocol Endpoint is not presented, vVol datastore creation will succeed, but then the datastore will enter an inaccessible status.

Though datastore creation through vCenter is a common task, it is covered herein since vVols are a new paradigm.

vVol Datastore Steps 1-4

Start by accessing the Storage icon in the Home page of the vSphere Client as shown previously in Figure 50. Then in Figure 55 below highlight the datacenter (step 1), select the Datastores tab (step 2) on the right menu, then from the ACTIONS menu (step 3), navigate to Storage -› New Datastore (step 4).

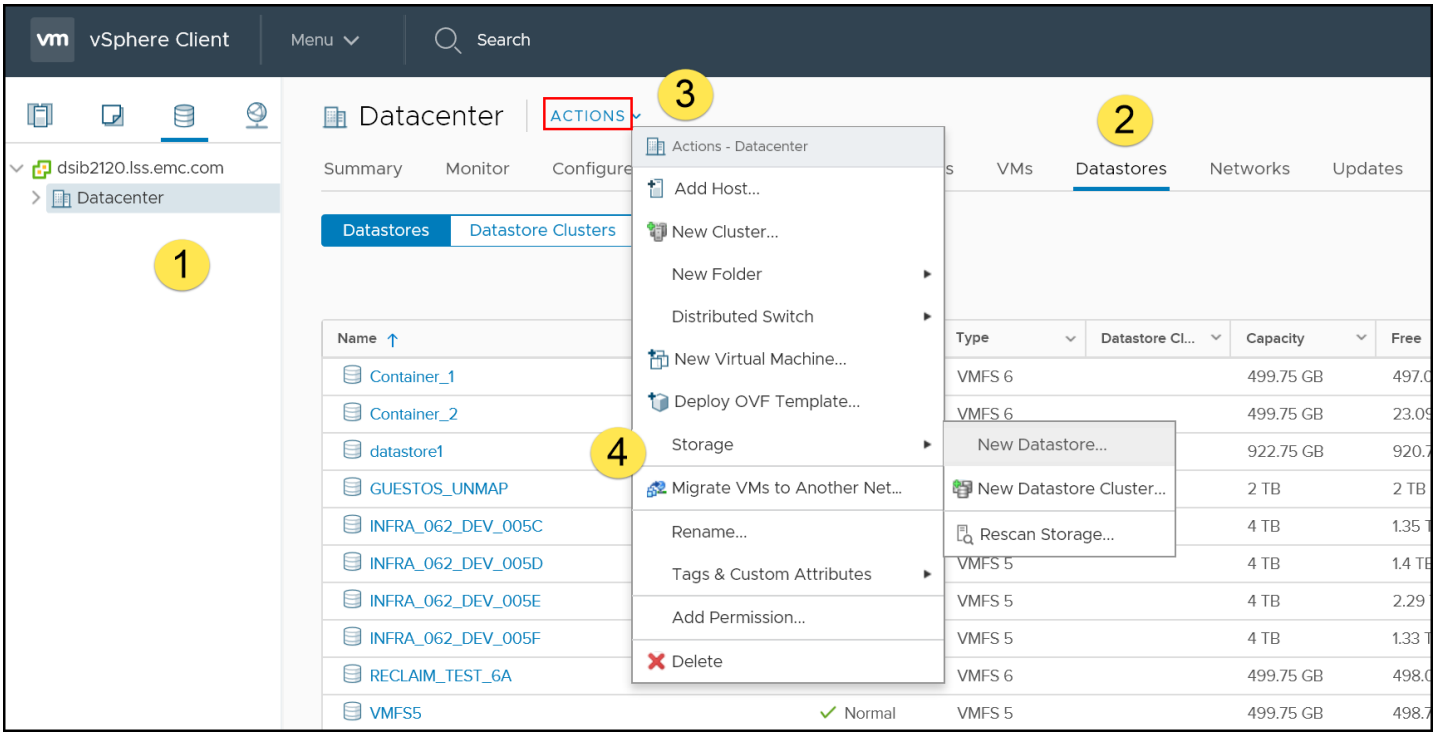

Figure 55. Creating a vVol datastore - steps 1-4

vVol Datastore Step 5

Step 5 prompts for the type of datastore. Select vVol[10] and Next.

[10] The vSphere thick client will not have this option and therefore cannot be used for this task.

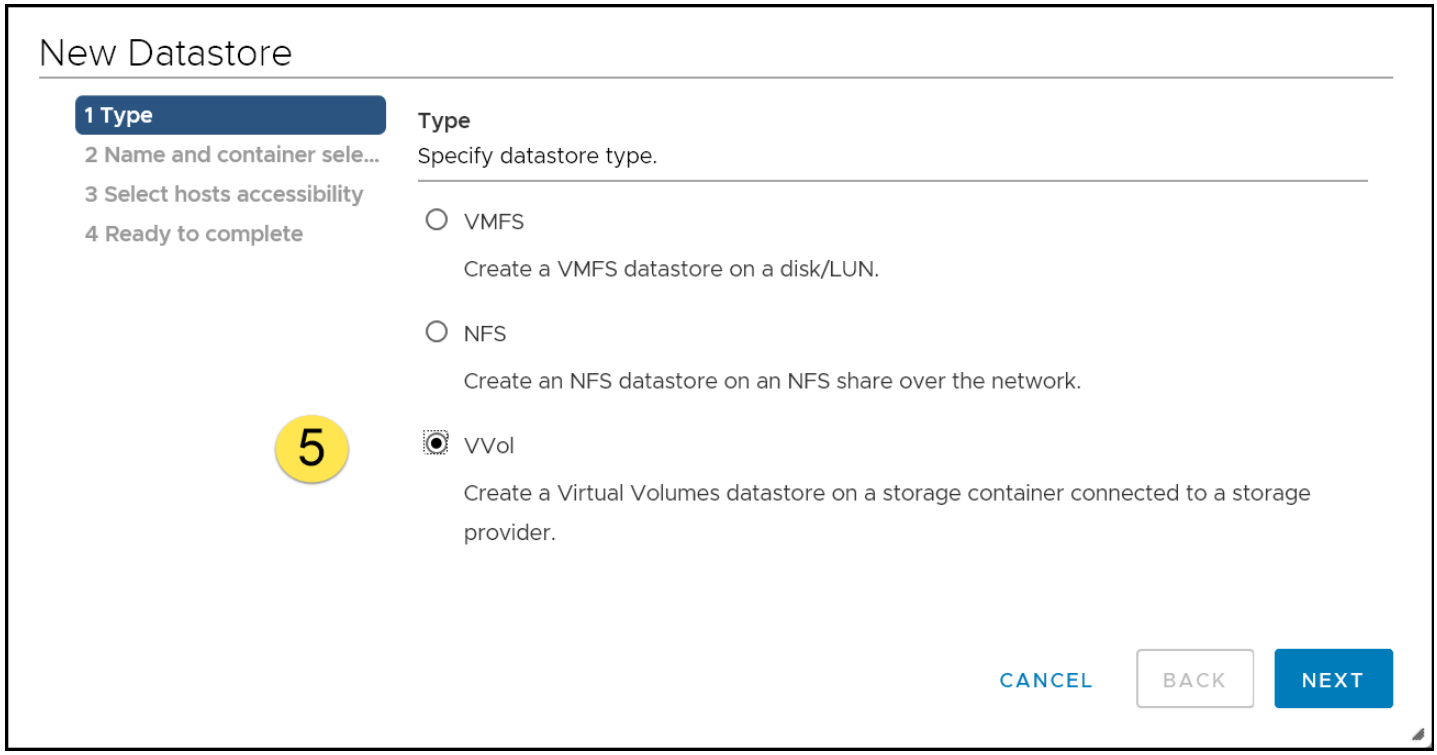

Figure 56. Creating a vVol datastore - step 5

vVol Datastore Step 6

Select a name for the datastore in step 6 and the storage container that is to be associated with that datastore. Recall that the vVol datastore is the vSphere representation of the VMAX storage container in the vCenter. Figure 57 shows the one storage container from which to choose. Note that the column "Maximum Disk Size" will always show as 16 TB. This refers to the maximum size of a single vVol that Dell EMC imposes. It has no association with the size of the storage container. The size of the container will not be visible until after datastore creation.

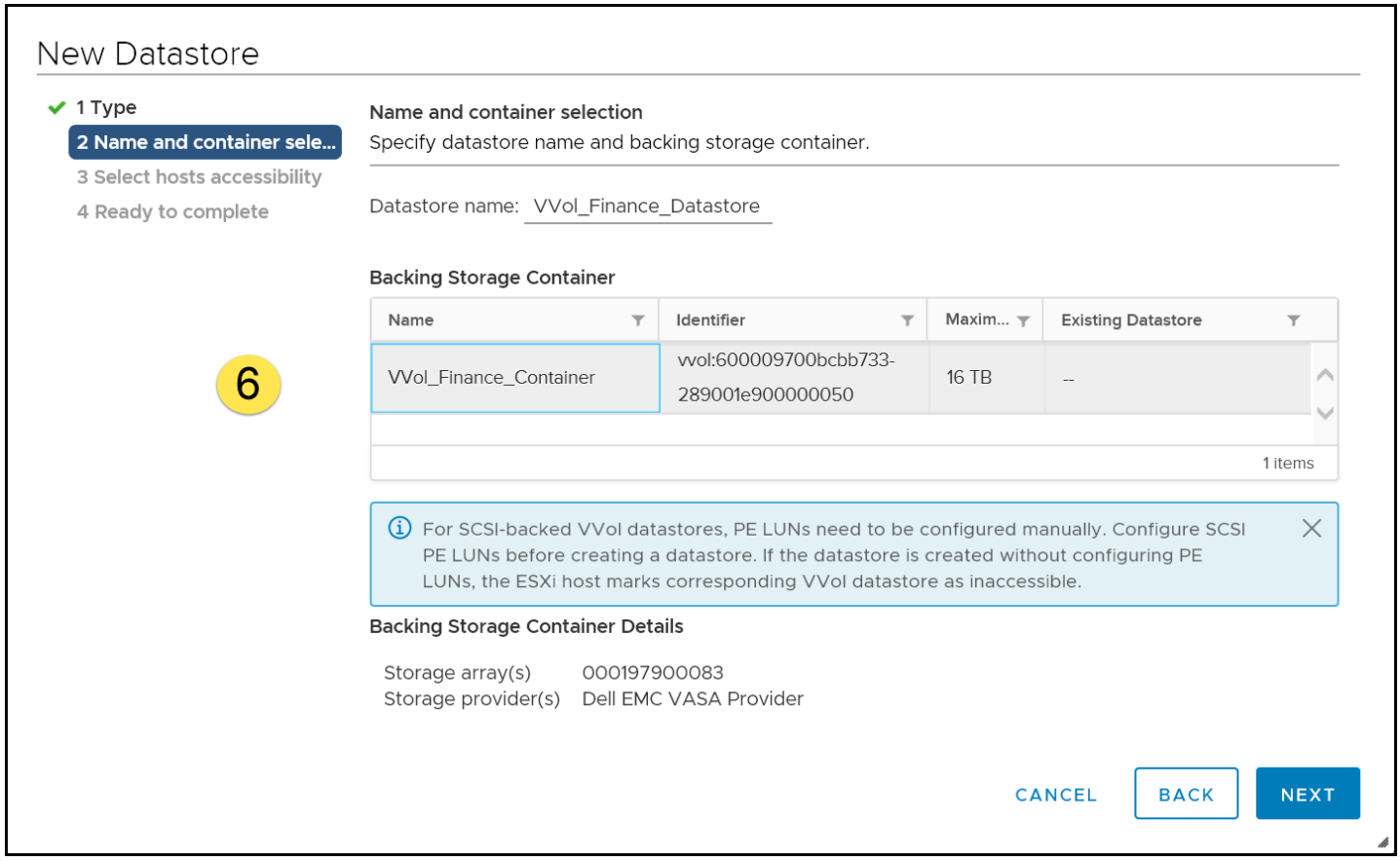

Figure 57. Creating a vVol datastore - step 6

vVol Datastore Step 7

In step 7 select on to which hosts the datastore should be mounted. In Figure 58 the one available host is chosen. When working with a cluster, all hosts will be shown. Be sure any selected hosts have a Protocol Endpoint.

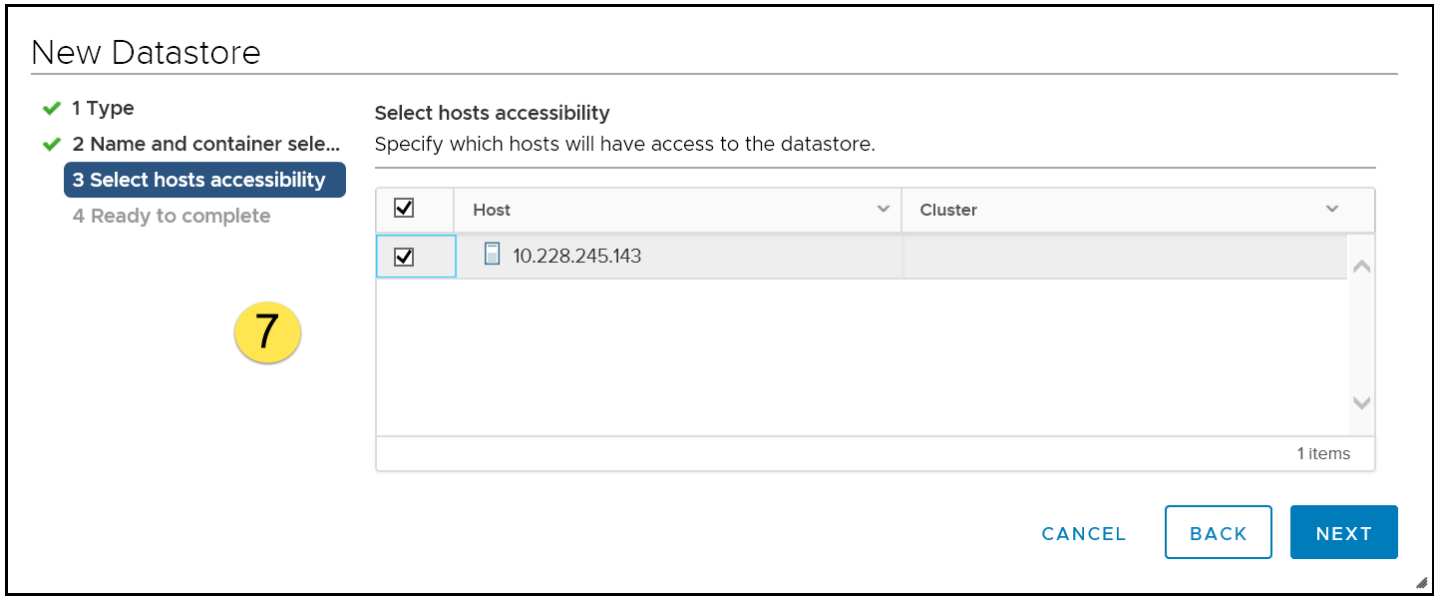

Figure 58. Creating a vVol datastore - step 7

vVol Datastore Step 8

In step 8 in Figure 59 review the chosen options and when ready select Finish. The datastore will be created and mounted to the host.

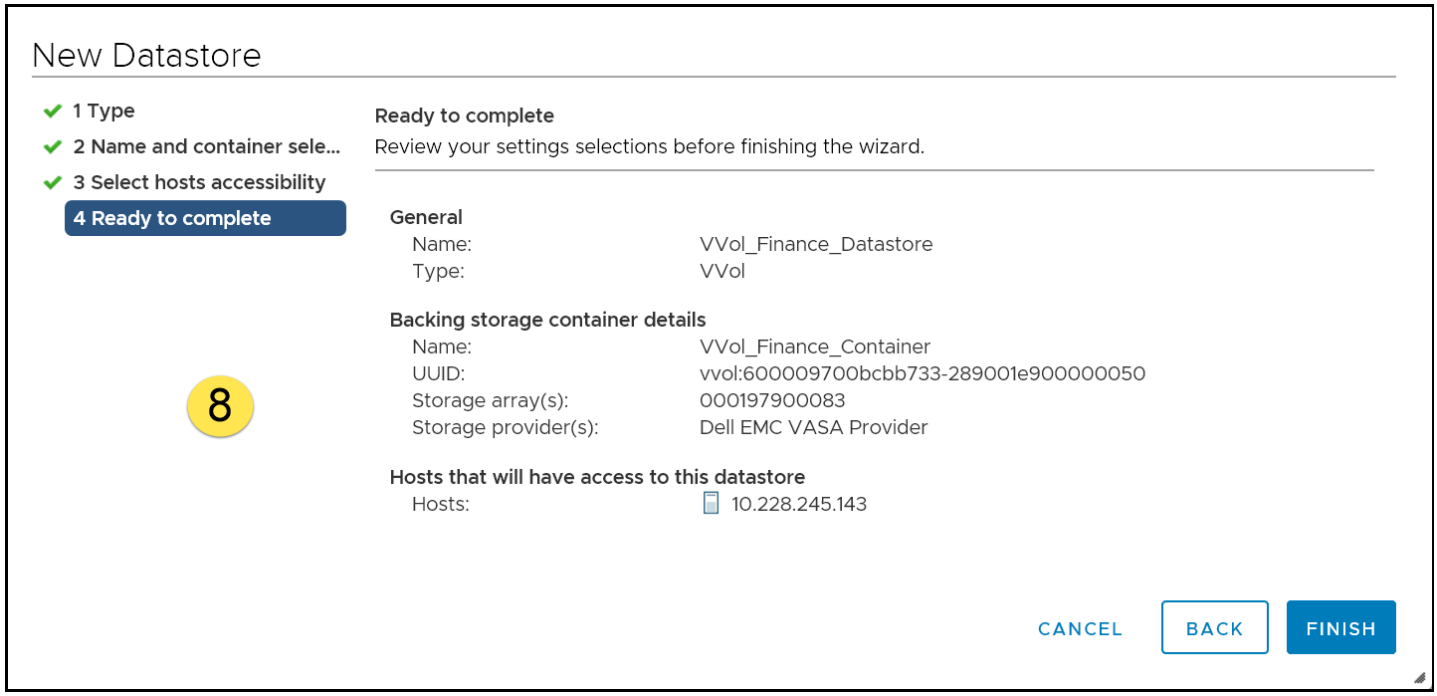

Figure 59. Creating a vVol datastore – capacity

Once created, the vVol datastore's storage capacity as viewed in vSphere (Figure 60) is the sum of the subscribed capacity for the storage container's storage resources. In this case there were two storage resources, one with 2 TB and one with 4 TB, totaling about 6 TB (rounding and metadata account for the discrepancies).

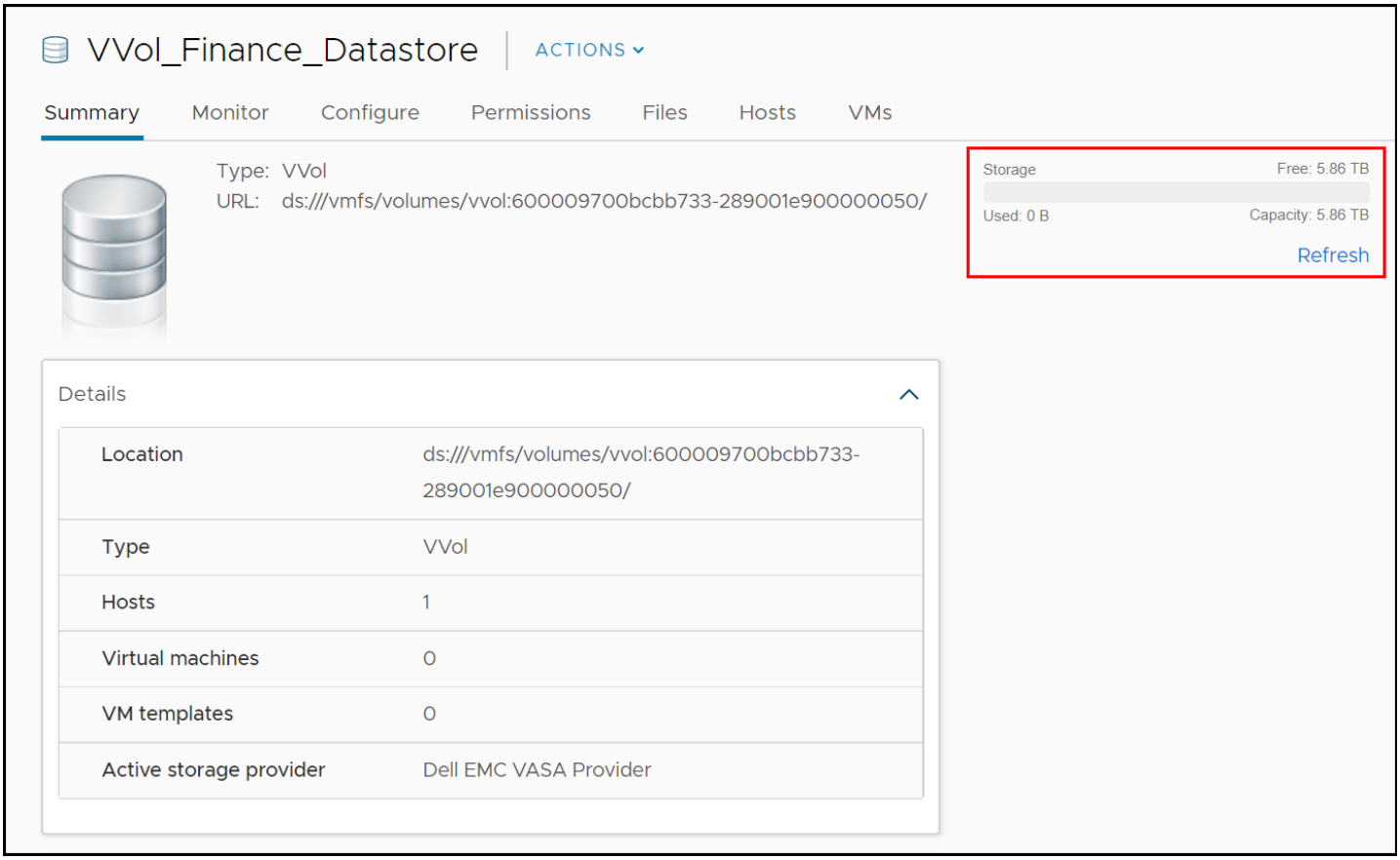

Figure 60. vVol datastore capacity

Due to a VMware bug which has not yet been resolved, the provisioned space of a vVol datastore that vCenter displays may show a value larger than the total capacity minus the free space. This number, however, has no bearing on functionality and can be ignored. As Dell EMC allocates space in the storage container for the total size of each vVol, the free space value is always accurate.

Modifying the Storage Container in Unisphere

If an existing storage resource in a storage container is changed – space added or removed – those changes will be reflected in vSphere upon refresh of the datastore. The following section provides an example.

At any time, a storage container may be modified by the storage administrator. A container may have a new storage resource added or removed or space added or removed to existing storage resources. In order to see the changes in the vCenter, one of two actions will be required. If a storage resource is added, it is necessary to rescan the VASA Provider in the vCenter. If an existing storage resource is modified, however, a simple refresh of the vVol datastore will show the changes.

SC Modification Step 1

From the VVol dashboard, Figure 61, select the Storage Containers icon.

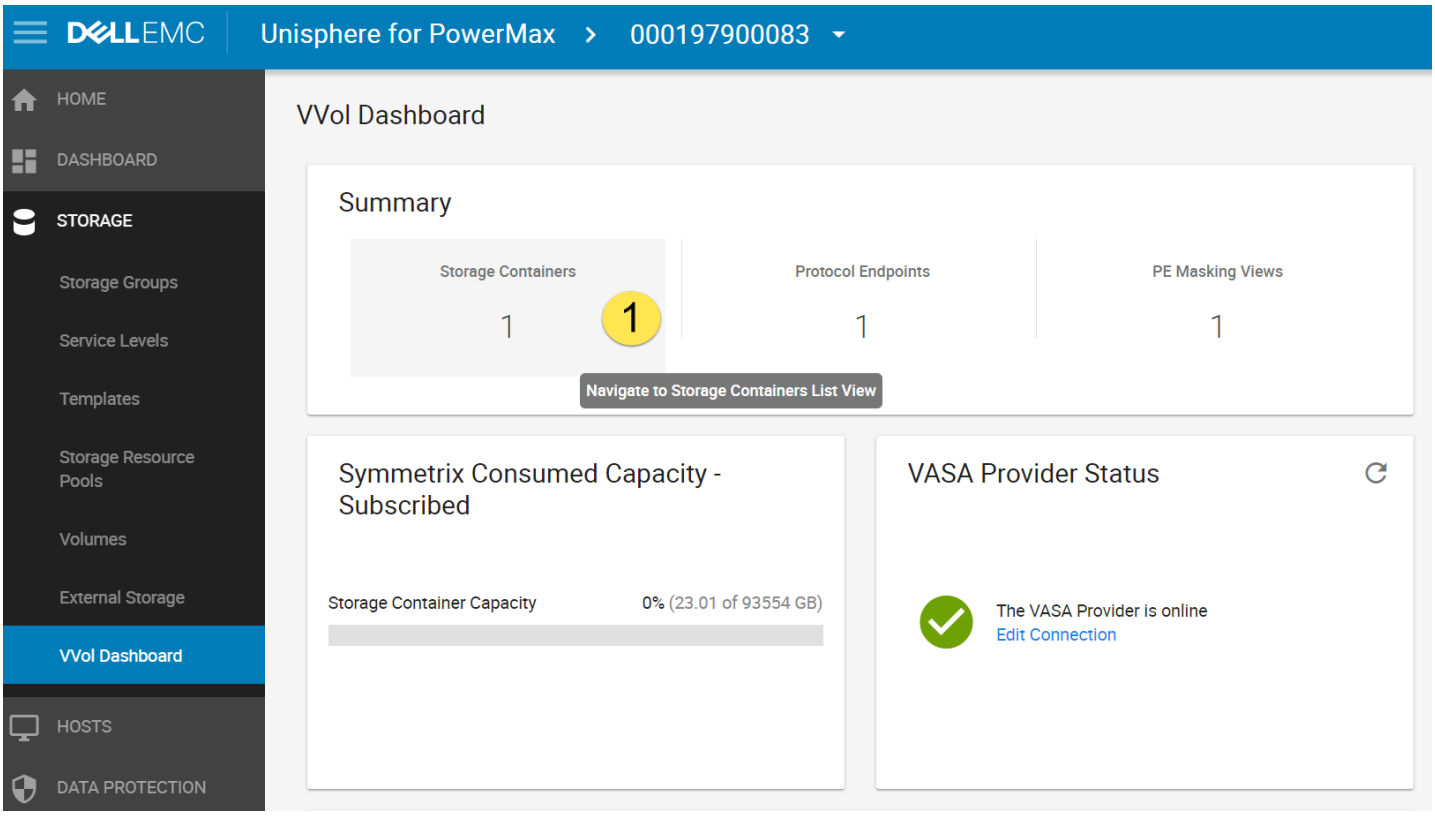

Figure 61. Modify Storage Container – step 1

SC Modification Step 2-3

In step 2 highlight the storage container and double-click to reveal the details in the panel on the right-hand side shown in Figure 62. Then in step 3 click on the hyperlink for the Storage Resources in the panel.

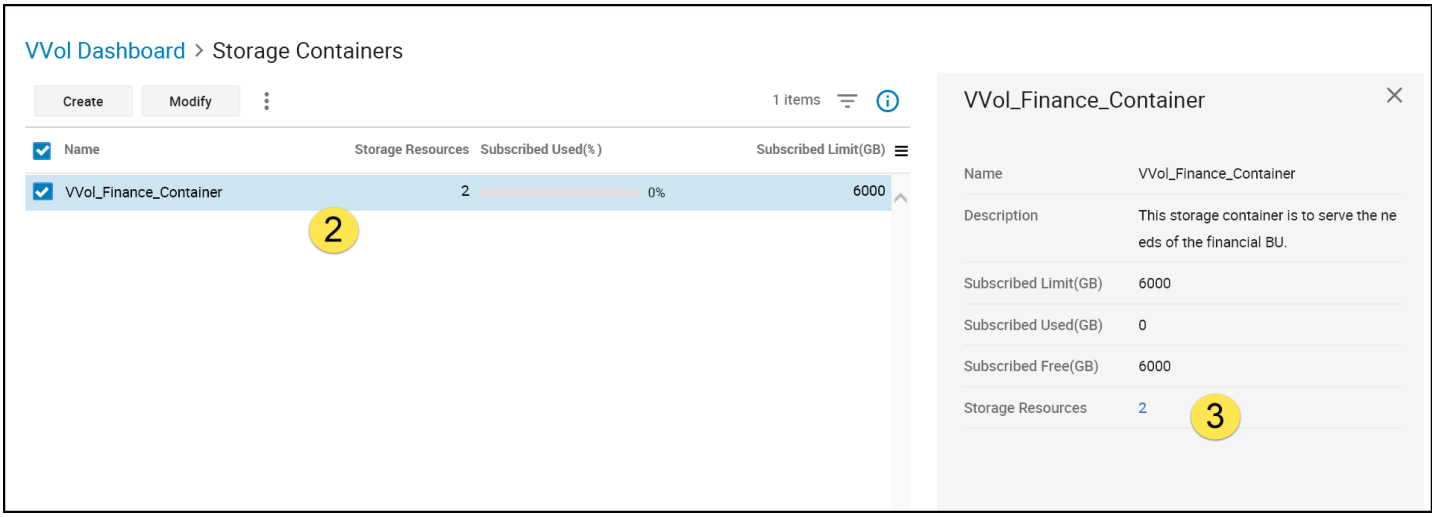

Figure 62. Modify Storage Container – step 2-3

SC Modification Step 4

In step 4 highlight one of the storage resources and select Modify. This is shown in Figure 63.

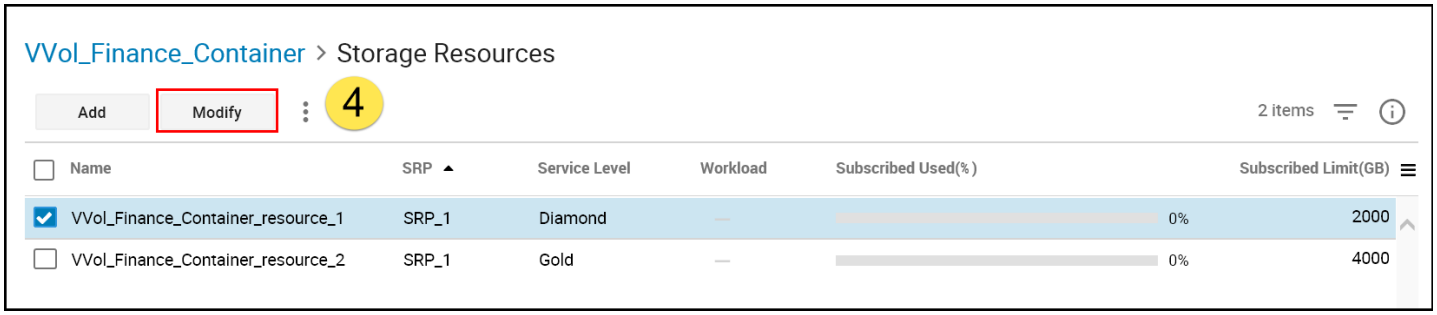

Figure 63. Modify Storage Container – step 4

SC Modification Step 5

In step 5 modify the Subscribed Limit to the new desired value. In this example the storage resource is increased from 2000 GB to 4000 GB as in Figure 64 and select OK.

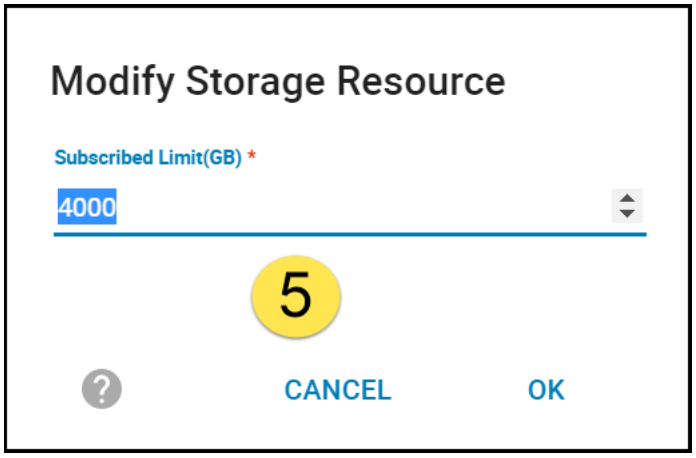

Figure 64. Modify Storage Container – step 5

The final result is shown in Figure 65.

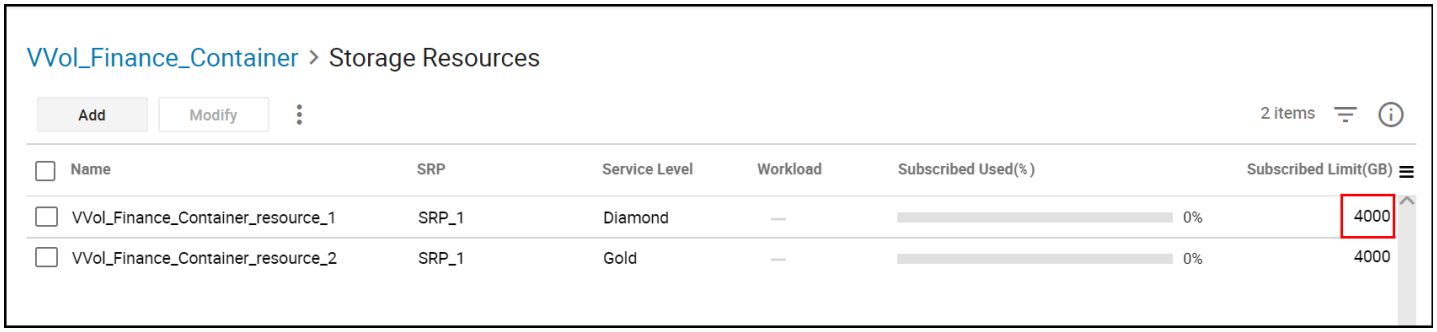

Figure 65. Modify Storage Container – completion

There are limitations as to what is possible when modifying the storage container. For instance, a storage resource cannot be resized below its used capacity. A storage resource also cannot be removed from a storage container while vVols are bound to it.

Recognizing new changes in vCenter

After the storage resource is modified, it is necessary to refresh the information in the vCenter. Simply select the REFRESH button in the vVol datastore screen to reflect the changes. This can be done from a variety of screens in the vCenter. Figure 66 demonstrates the change in the vVol datastore capacity when 2000 GB is added to the storage resource.

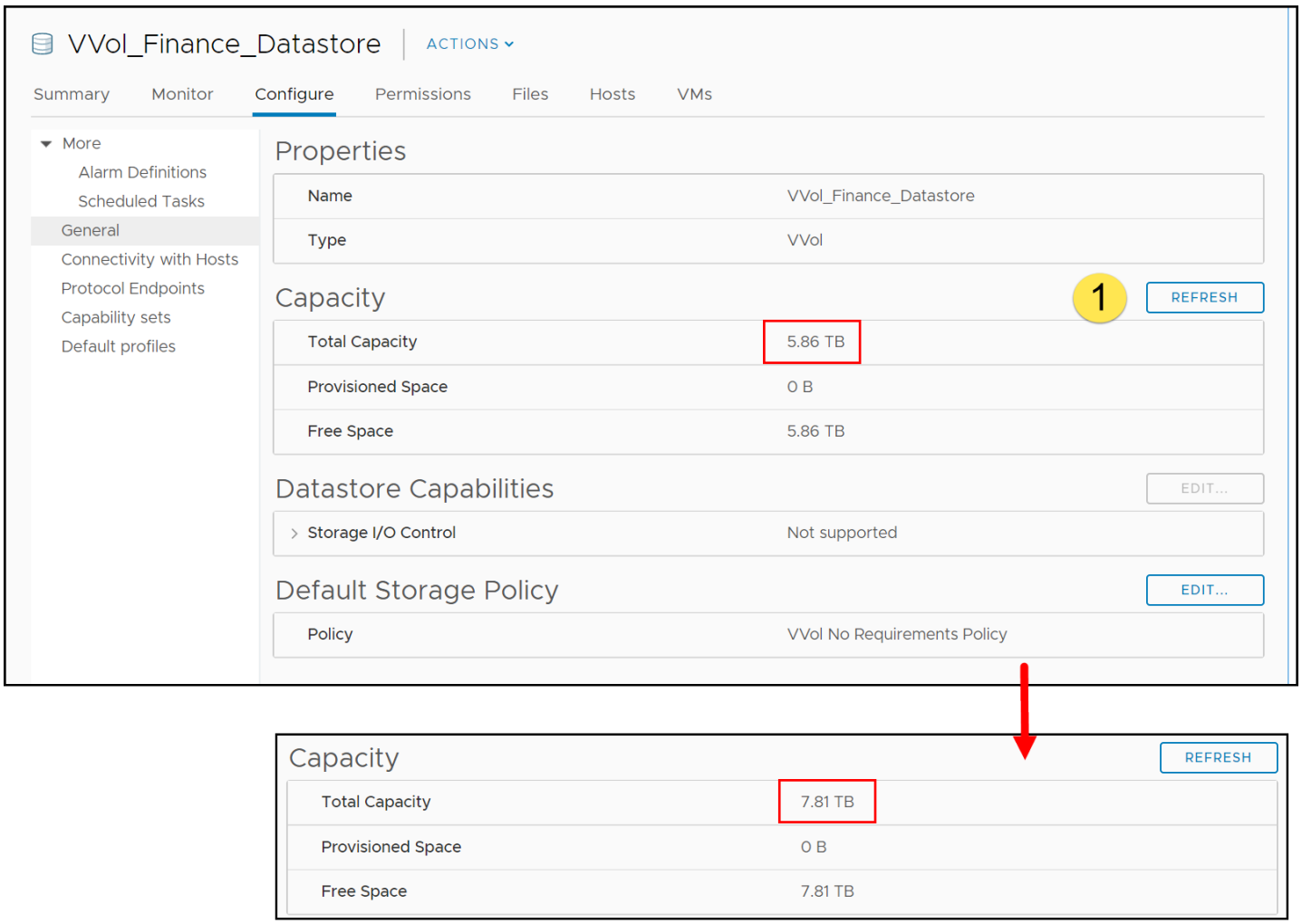

Figure 66. Refresh vVol datastore

If, however, a storage resource is added to the container rather than an existing one modified, as in Figure 67, the VASA Provider must be rescanned.

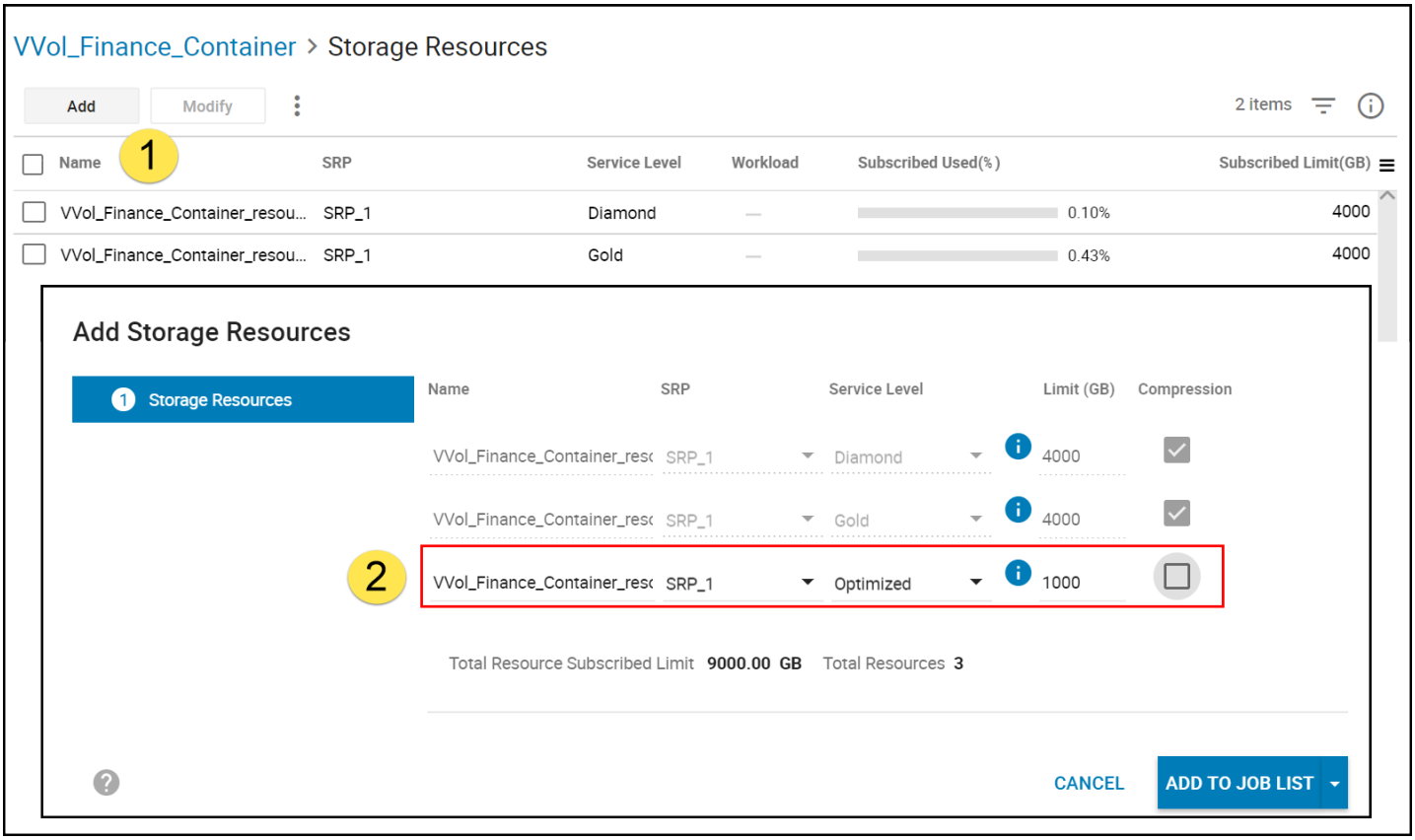

Figure 67. Adding a storage resource to a storage container

To do this, navigate to the Storage Providers screen as explained in the section “vVols support the use of VMware Storage IO Control at the Storage Policy level, providing a way to limit IO to virtual machines. Figure 49 demonstrates the functionality.

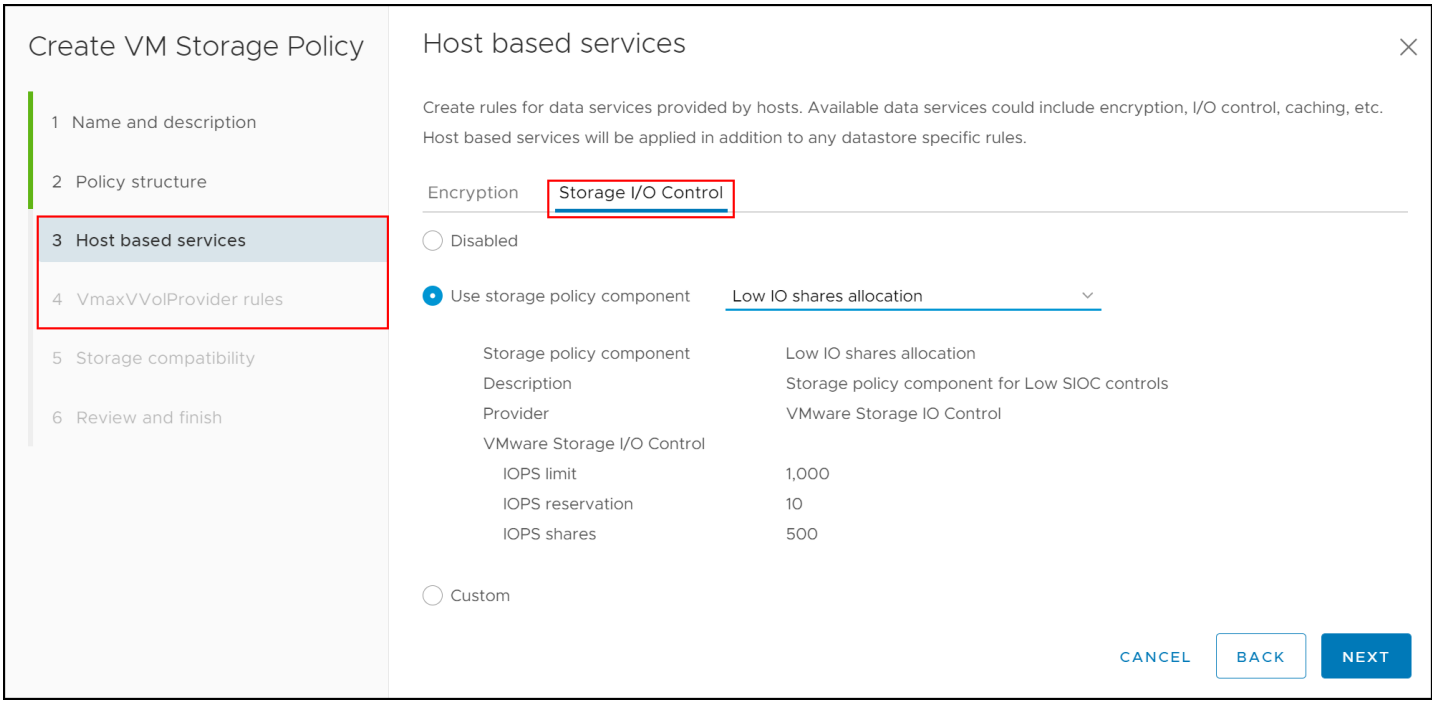

Figure 49. Storage I/O Control with vVols

Registering the VASA Provider in vCenter." Then highlight the Dell EMC VASA Provider in step 1 and select the Rescan option in step 2 as demonstrated in Figure 68.

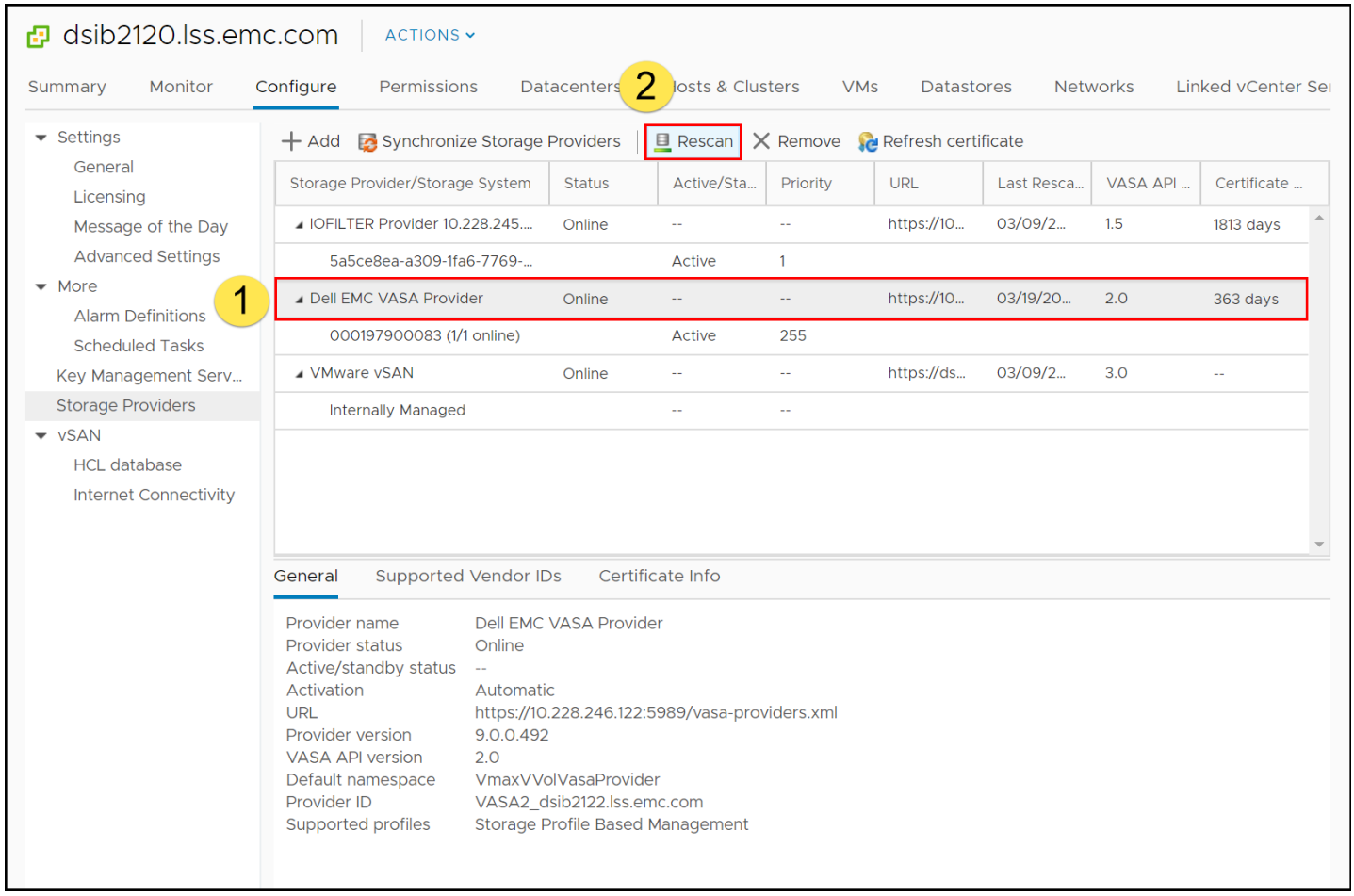

Figure 68. Rescan the VASA Provider

Once rescanned, the new storage resource will display under the storage capabilities of the vVol datastore shown below in Figure 69.

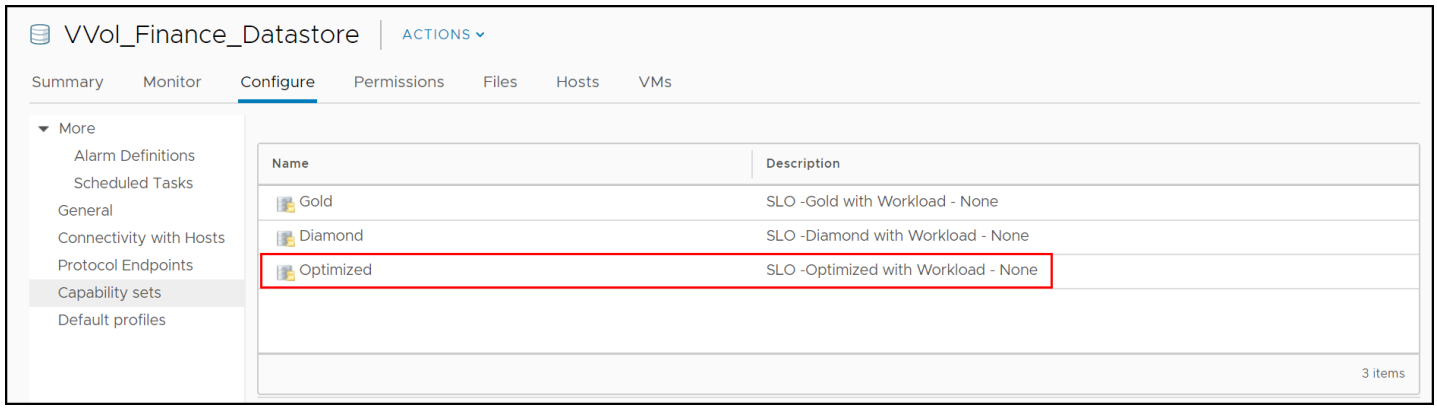

Figure 69. New storage resource displayed after rescan

Creating a VM Storage Policy for vVols

VMware utilizes Storage Policy Based Management in conjunction with vVols. The VMAX advertises its capabilities to vSphere. The user creates policies that map to those capabilities so that when the user provisions a VM, a policy can be selected that will filter the datastores so that the appropriate one is selected. The capabilities are passed to the VASA Provider so that in the case of the VMAX and PowerMax, the vVols are created with the proper Service Level Objective (SLO). Table 1 lists the available SLs for VMAX and PowerMax while Figure 70 and Figure 71 show a graphical representation of them in Unisphere. Note that the expected average response time of an SL in VMAX (VMAX3) is different than PowerMax. Furthermore, while VMAX uses different tiers of disks to achieve the response time of a particular SL, the PowerMax is an all flash array, and therefore throttles IO to achieve the desire response time.

Table 1. VMAX and PowerMax storage capabilities in vSphere

SystemLabel					
Optimized	Bronze	Silver	Gold	Platinum	Diamond
	Bronze+OLTP	Silver+OLTP	Gold+OLTP	Platinum+OLTP	Diamond+OLTP
	Bronze+OLTP_REP	Silver+OLTP_REP	Gold+OLTP_REP	Platinum+OLTP_REP	Diamond+OLTP_REP
	Bronze+DSS	Silver+DSS	Gold+DSS	Platinum+DSS	Diamond+DSS
	Bronze+DSS_REP	Silver+DSS_REP	Gold+DSS_REP	Platinum+DSS_REP	Diamond+DSS_REP

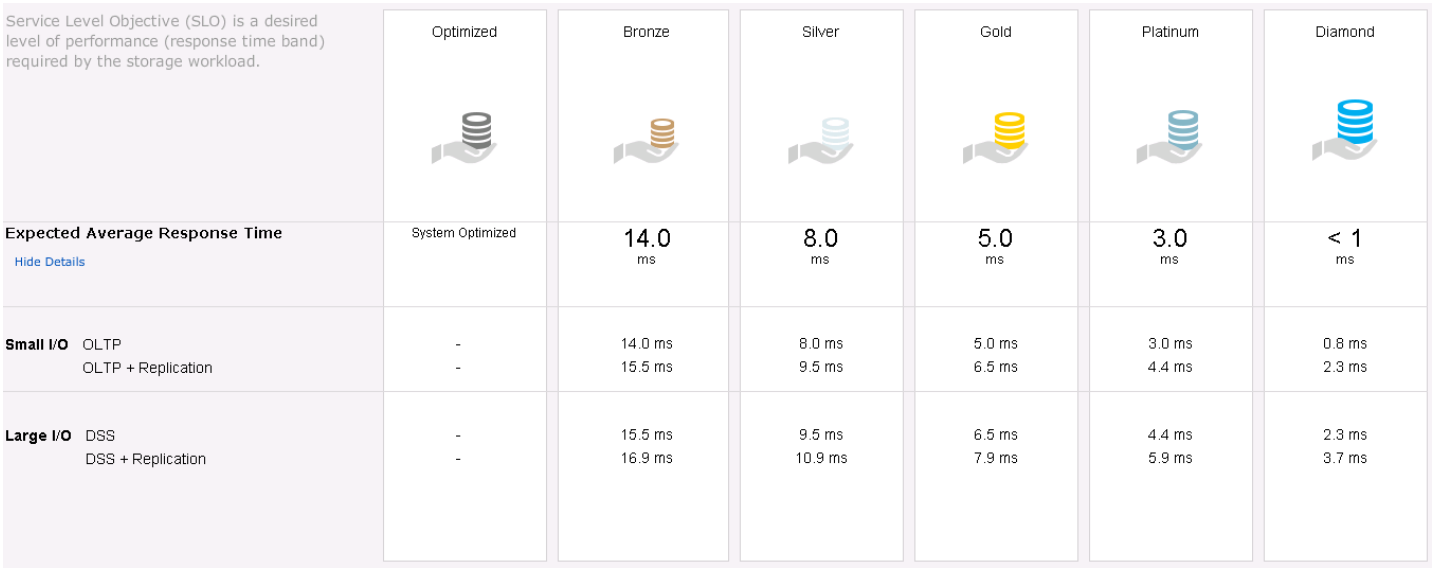

Figure 70. VMAX storage capabilities in Unisphere for VMAX

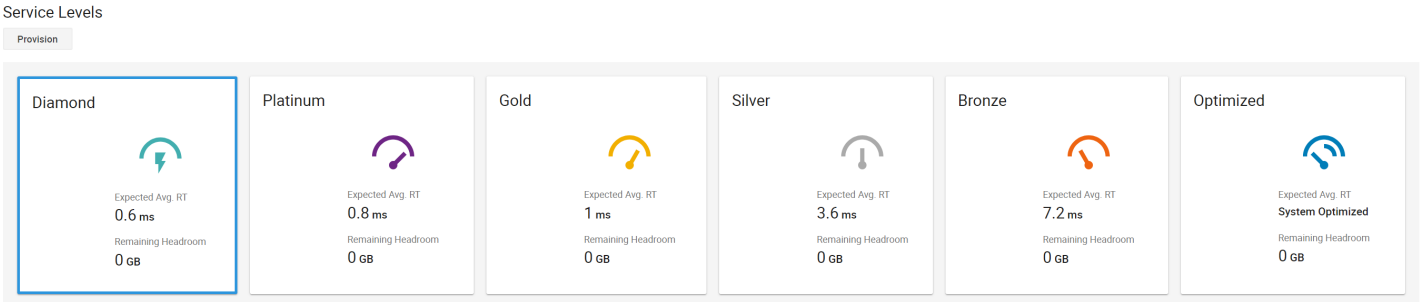

Figure 71. PowerMax storage capabilities in Unisphere for PowerMax

The following provides a step-by-step process for creating a VM storage policy in vSphere 6.7. As the wizard is different in vSphere 6.5, some screens are included for reference. A storage policy is not required in order to use vVol storage, however any VMs created in a vVol datastore without using a storage policy will be assigned the SL with the highest response time targets.

When using a VMAX All Flash array (AFA) there is only a single SL, Diamond, and three workload types: OLTP, DSS, and None (default). Therefore, creating a VM Storage Policy is optional unless the workload specification is desired as all vVols will be created with a Diamond SL when selecting a vVol datastore. Note also that the PowerMax does not use a workload specification.

VM Storage Policy Step 1

Start by accessing the VM Storage Policies icon in the Home page of the vSphere Client as shown in Figure 72.

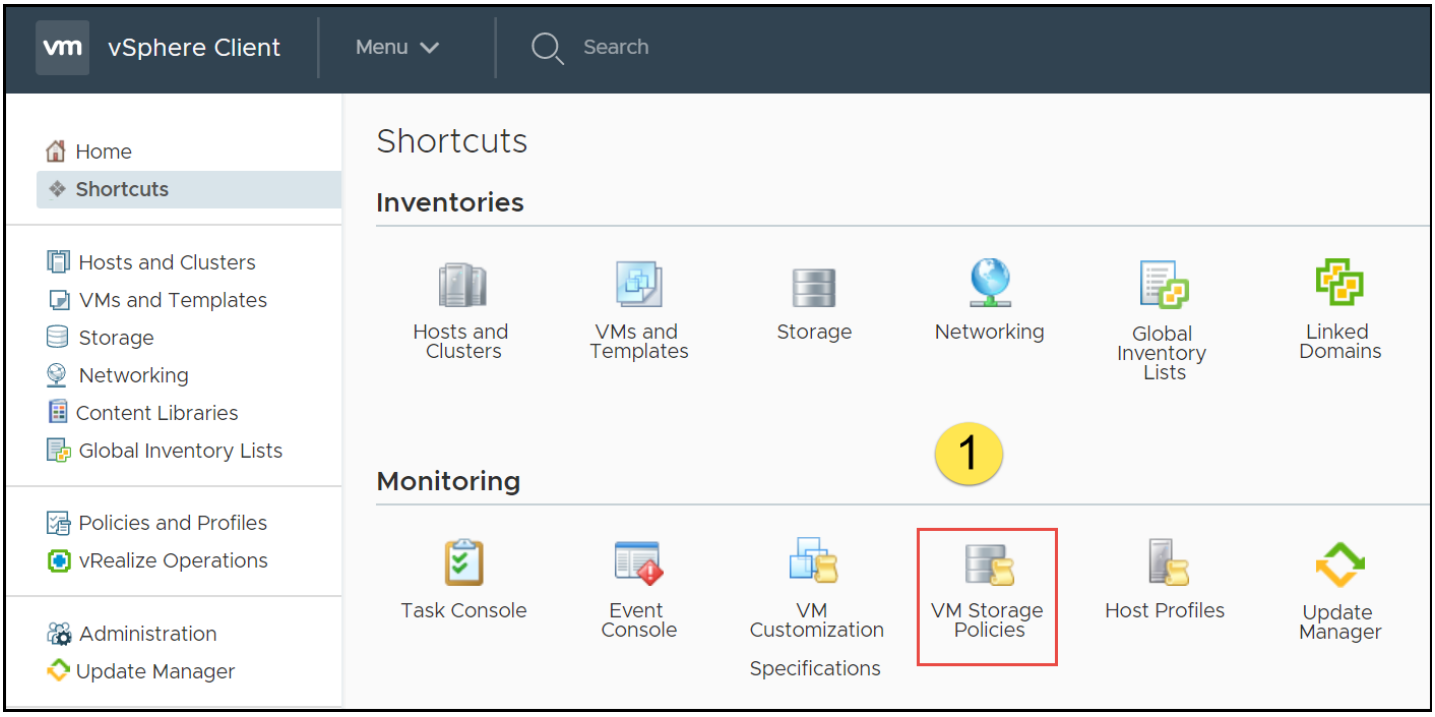

Figure 72. Creating a VM Storage Policy - step 1

VM Storage Policy Step 2

Next in step 2 in Figure 73 select the icon Create VM Storage Policy to create a new VM storage policy.

Figure 73. Creating a VM Storage Policy - step 2

VM Storage Policy Step 3

Step 3 (step 1 of the wizard) formally starts the wizard. If the environment shares a Platform Service Controller then begin by selecting the appropriate vCenter. Enter a name for the policy, preferably one that reflects the capabilities that will be associated with the policy as this is the name the VMware user will see. Finally, if desired enter a description. An example is shown in Figure 74 for vSphere 6.7 and Figure 75 for vSphere 6.5.

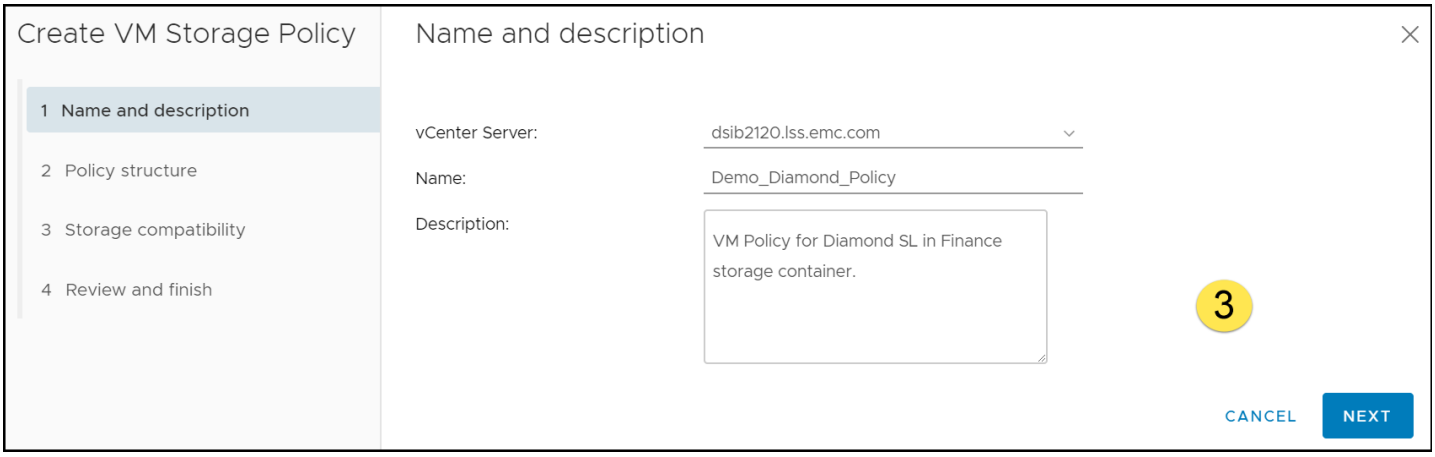

Figure 74. Creating a VM Storage Policy - step 3

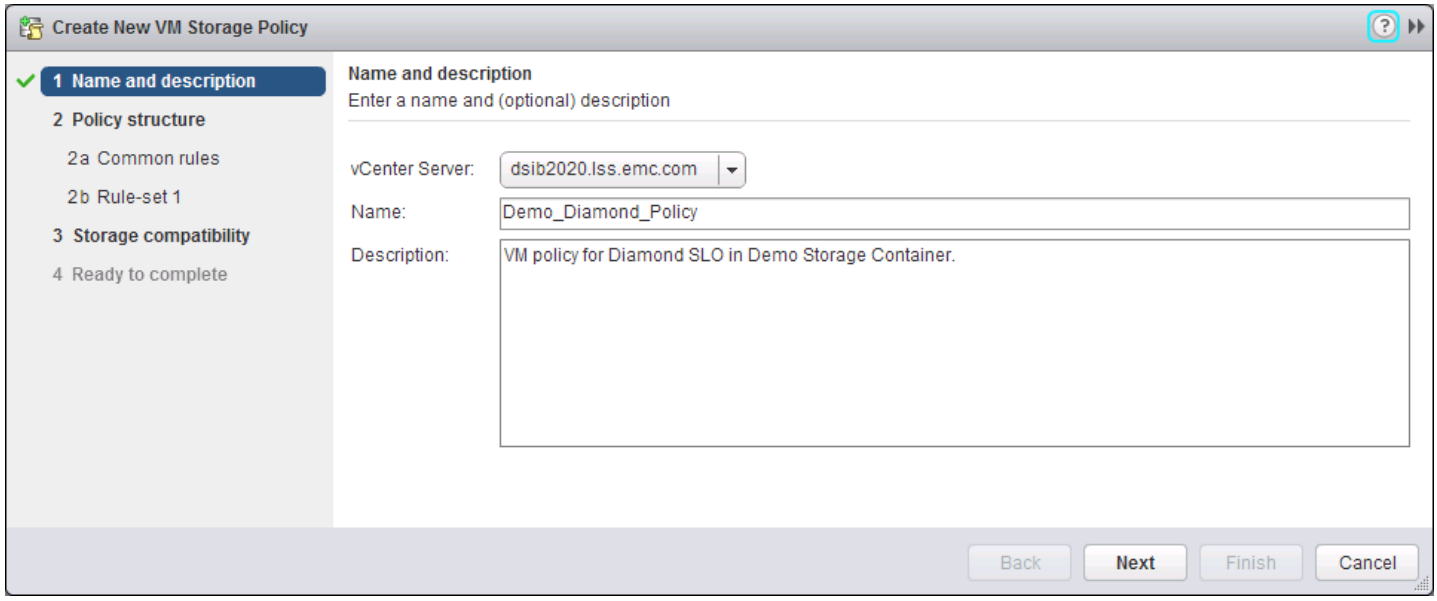

Figure 75. Creating a VM Storage Policy - step 3 (vSphere 6.5)

VM Storage Policy Step 4

Step 4 is the Policy structure definition. It covers adding host-based rules, like Storage IO Control, as well as assigning the Rule Set.

The Dell EMC VASA Provider 2.0 advertises two different data services:

- VmaxvVolProvider
- VmaxvVolVasaProvider.VASA10

The difference between the two services is straightforward – the first supports VASA 2.0 and vVol functionality while the second supports the older VASA 1 capabilities. For this task, the VmaxvVolProvider is used. VASA 1 capability is discussed in the section Creating a VM Storage Policy for VMFS. Therefore, select the VmaxvVolProvider as demonstrated in Figure 76.

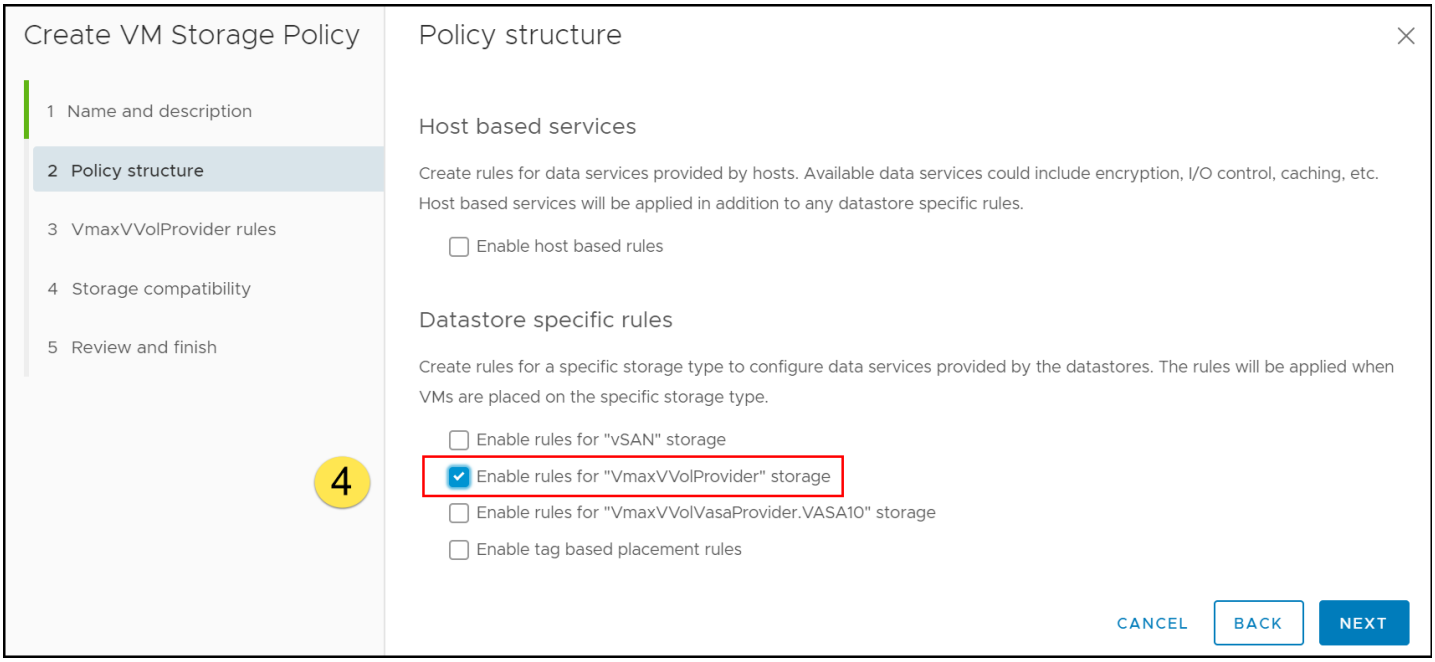

Figure 76. Creating a VM Storage Policy - step 4 (vSphere 6.7+)

In vSphere 6.5, the Policy structure is broken out into three screens, rather than one. The first screen simply describes what Policy structure is in vSphere 6.5. It is included here in Figure 77 for reference and continuity.

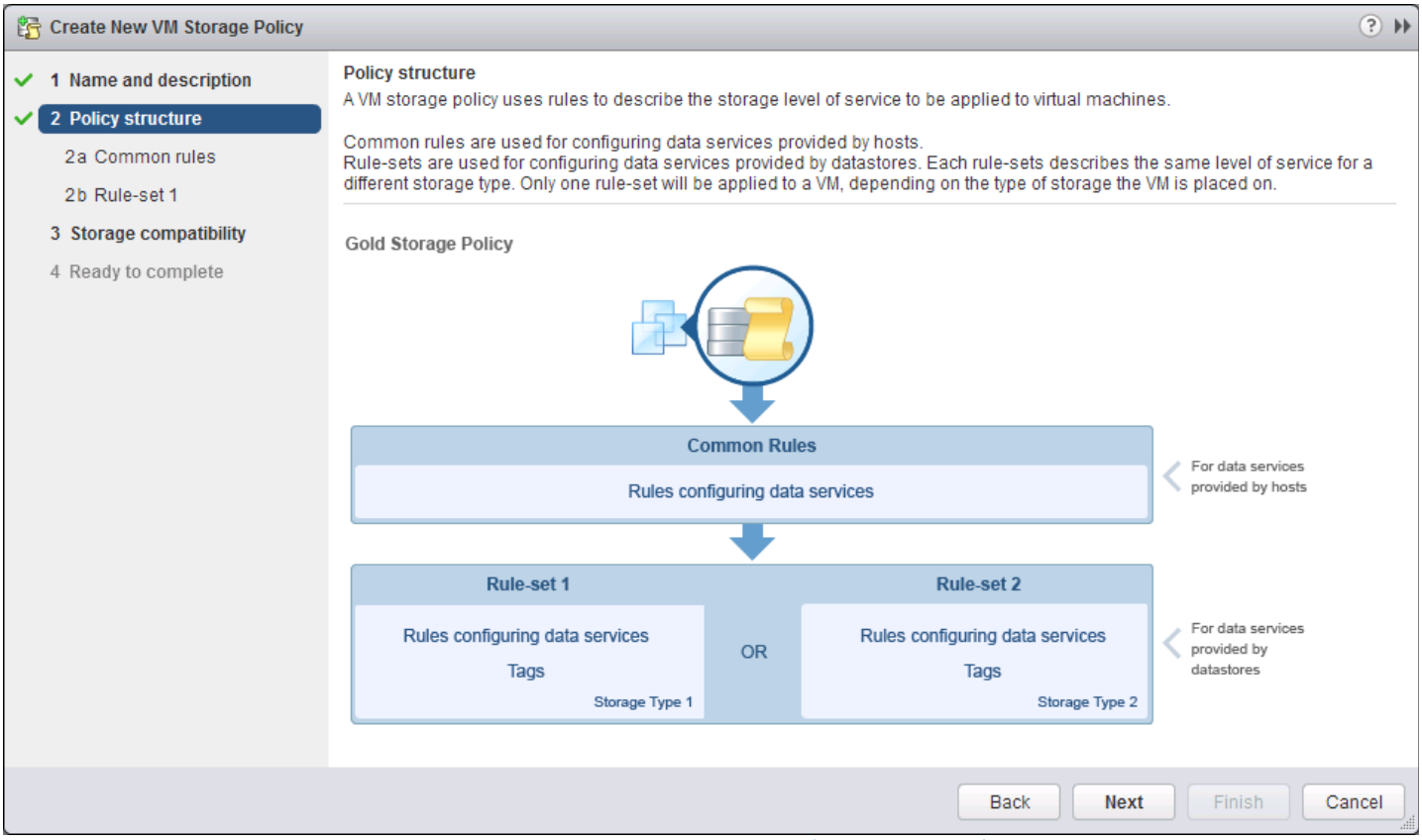

Figure 77. Creating a VM Storage Policy - step 4 (vSphere 6.5)

In the second screen in Figure 78, the common or host rules can be added such as Encryption.

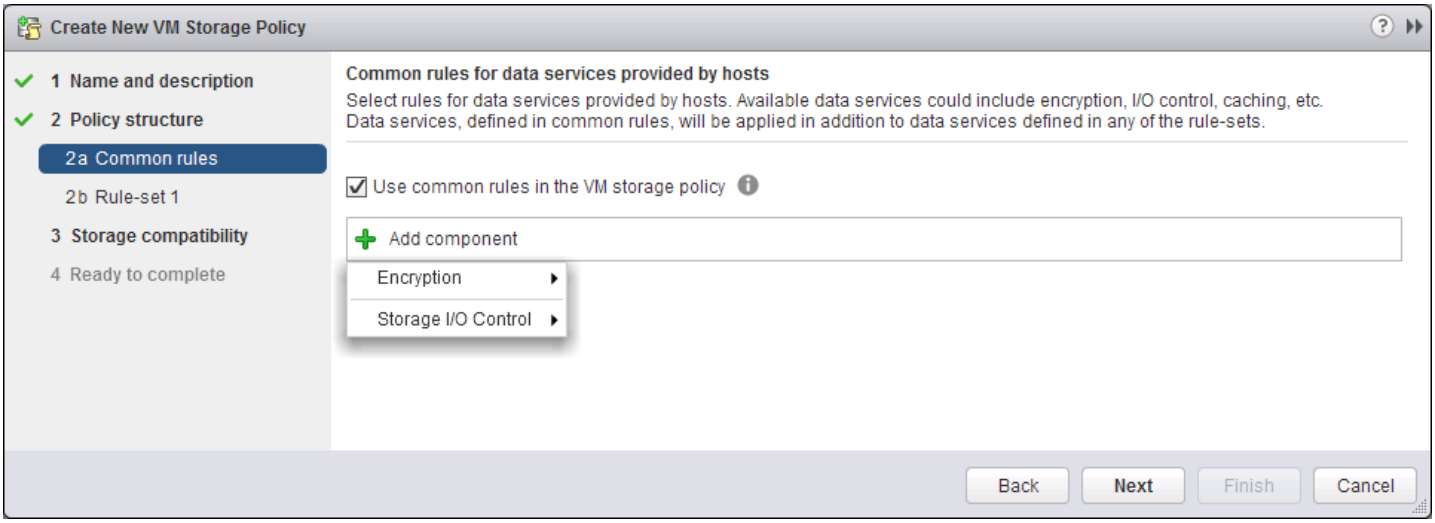

Figure 78. Creating a VM Storage Policy - step 4 (vSphere 6.5)

In the final screen in Figure 79, the rule-set is selected. It is the same as in vSphere 6.7, VmaxvVolProvider.

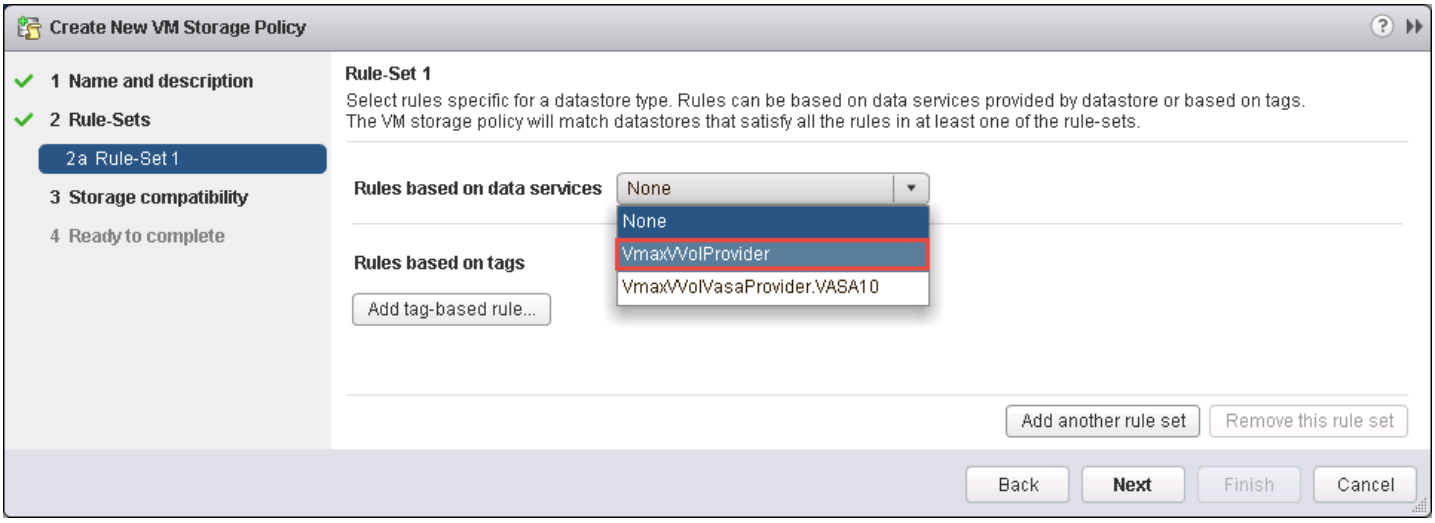

Figure 79. Creating a VM Storage Policy - step 4 (vSphere 6.5)

VM Storage Policy Step 5

Once the data service is selected, the advertised capabilities may be added as rules for the policy. Through the VASA Provider the VMAX presents the Service Level Objective as the capability. It is comprised of two rules, from a VMware perspective: Performance Index and Workload Hints. Start by selecting Performance Index as in Figure 80 and Figure 81.

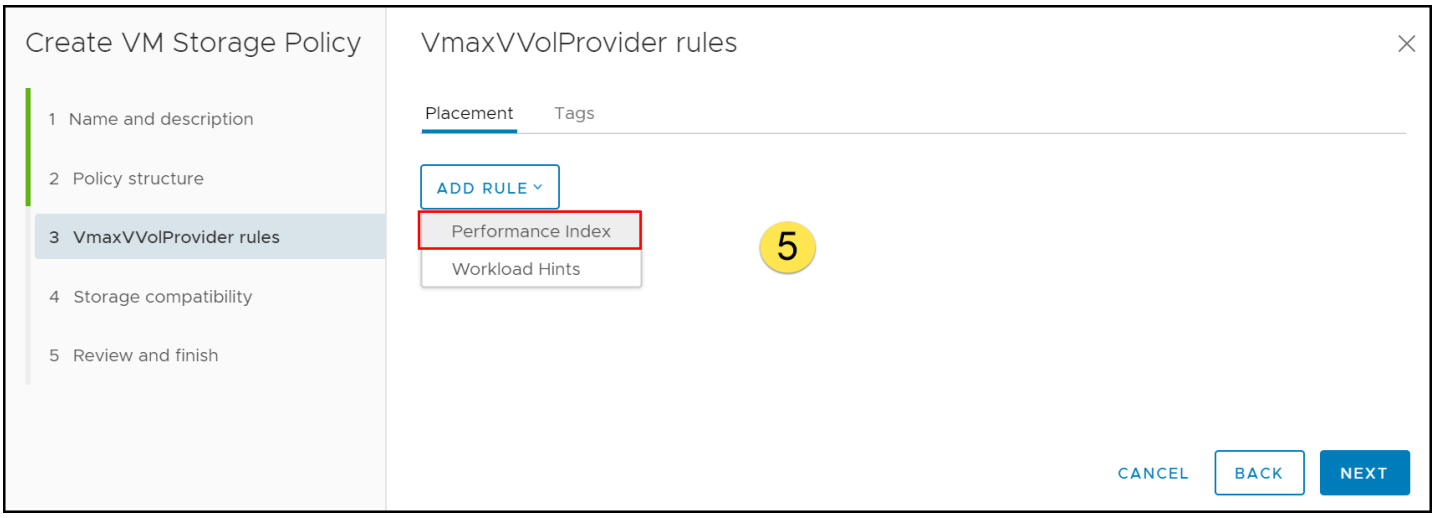

Figure 80. Creating a VM Storage Policy - step 5 (vSphere 6.7+)

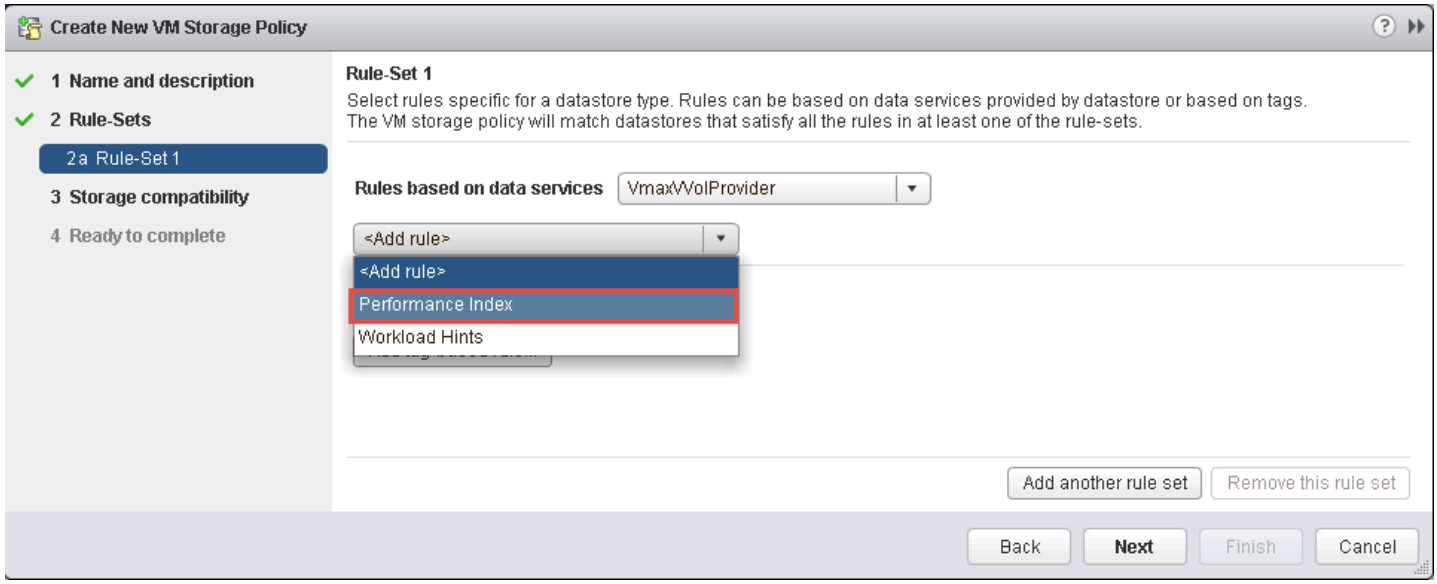

Figure 81. Creating a VM Storage Policy - step 5 (vSphere 6.5)

VM Storage Policy Step 6

In step 6, Figure 82, one can see that the Performance Index for VMAX will be translated into "Service Level" and all the SLs available on the box will be shown in a drop-down list. Select an SL that maps to a storage resource available in the storage container on the VMAX.

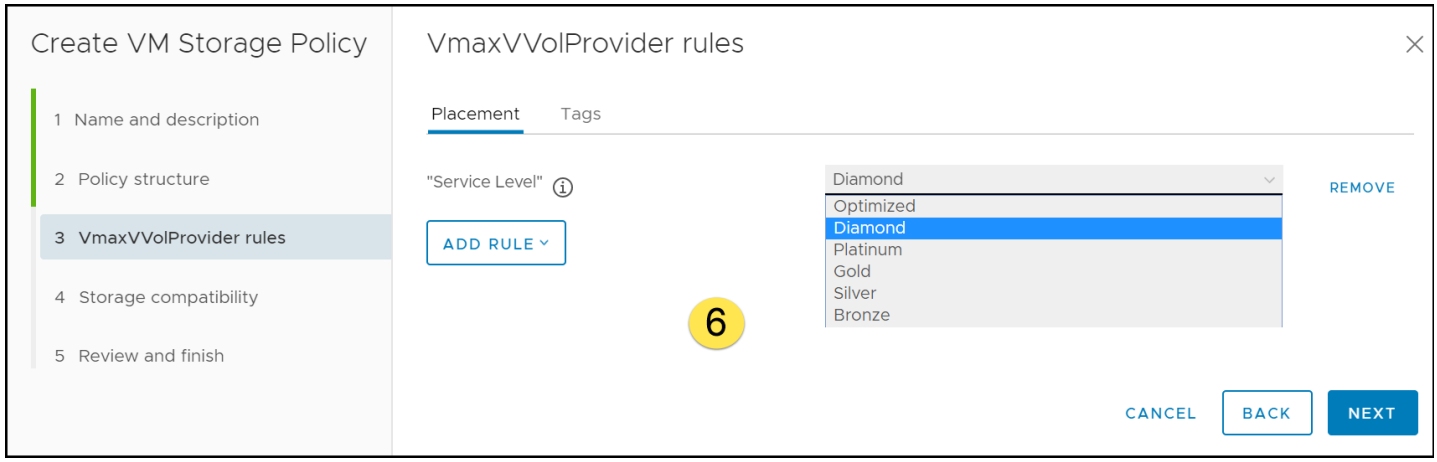

Figure 82. Creating a VM Storage Policy - step 6 (vSphere 6.7+)

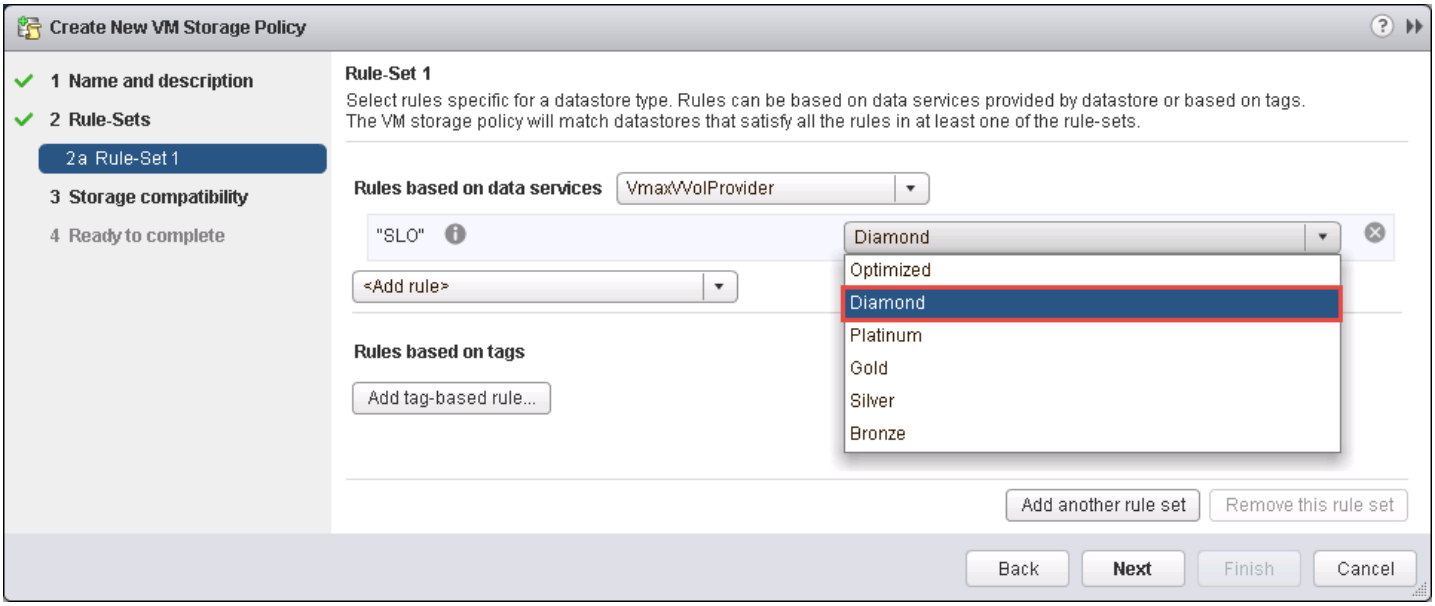

Figure 83. Creating a VM Storage Policy - step 6 (vSphere 6.5)

VM Storage Policy Step 7

Step 7 is not a required step since it is used to specify a Workload Hint. If the storage resource has a workload associated with it (OLTP, DSS), then select this option as in Figure 84 and Figure 85.

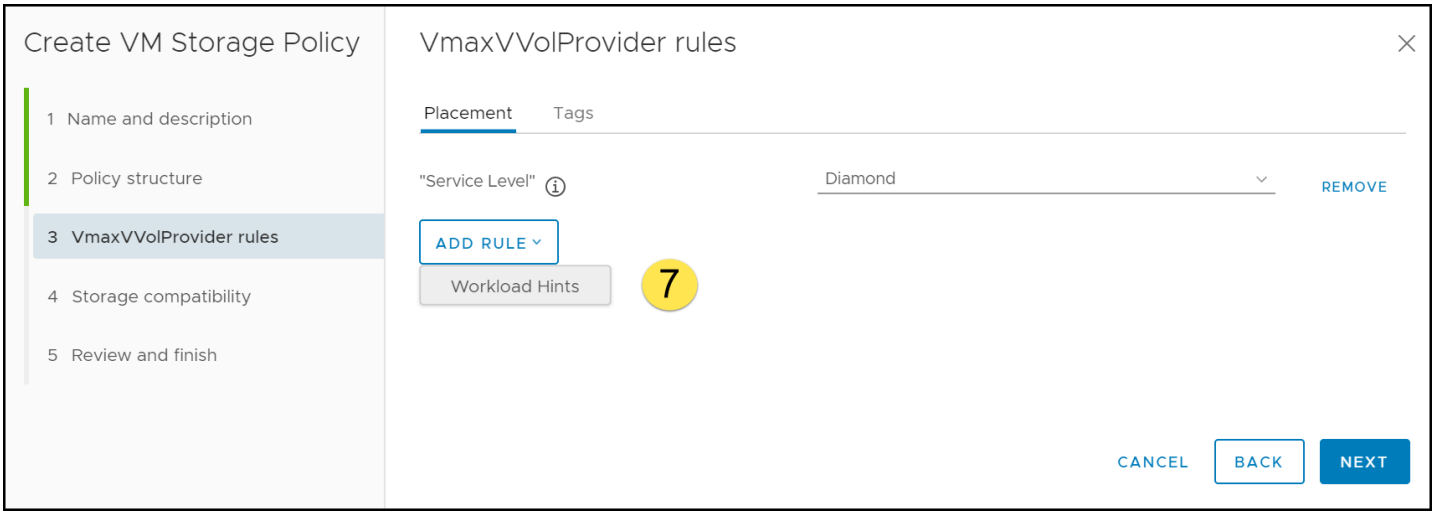

Figure 84. Creating a VM Storage Policy - step 7 (vSphere 6.7+)

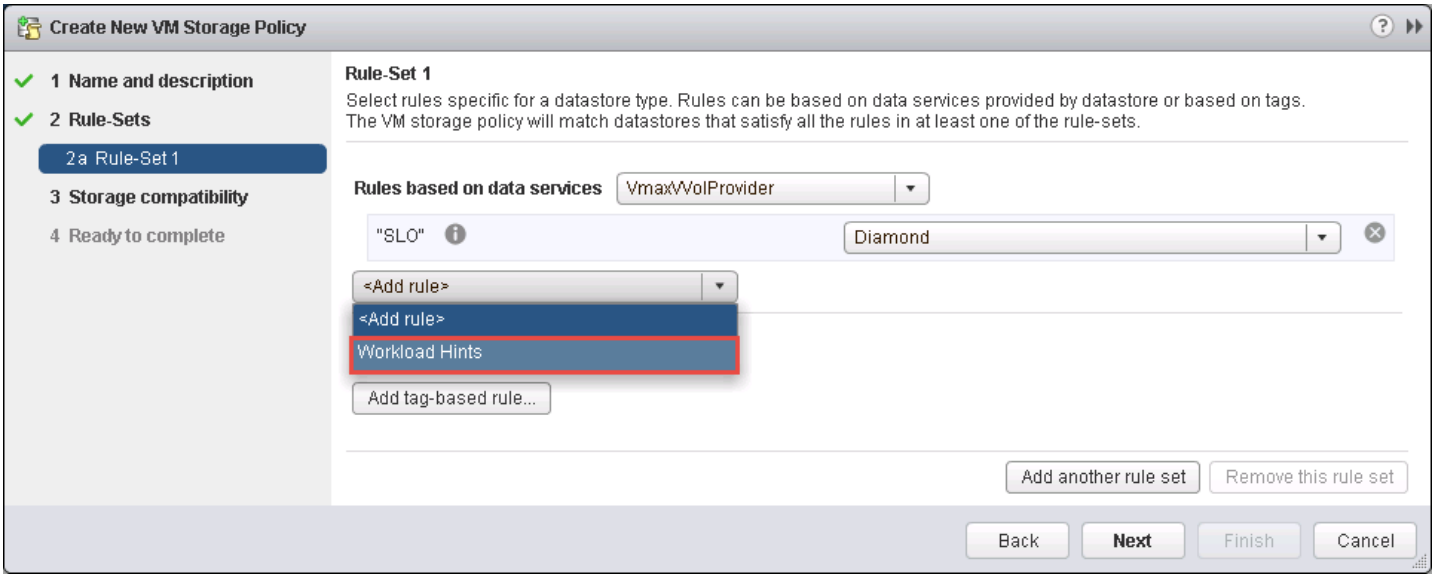

Figure 85. Creating a VM Storage Policy - step 7 (vSphere 6.5)

VM Storage Policy Step 8

Choose the Workload – OLTP, DSS or None – in Figure 86 or Figure 87. The reason "None" is included as a workload (since it would seem implied) is if it is critical to only use a storage resource if it does not have a workload, the user must select None. If the user only selects an SLO with no workload, but the storage container contains a storage resource with said SLO and a workload (OLTP or DSS), the vVol datastore will be compatible and the vVol will be created in that storage resource since it matches the generic SLO.

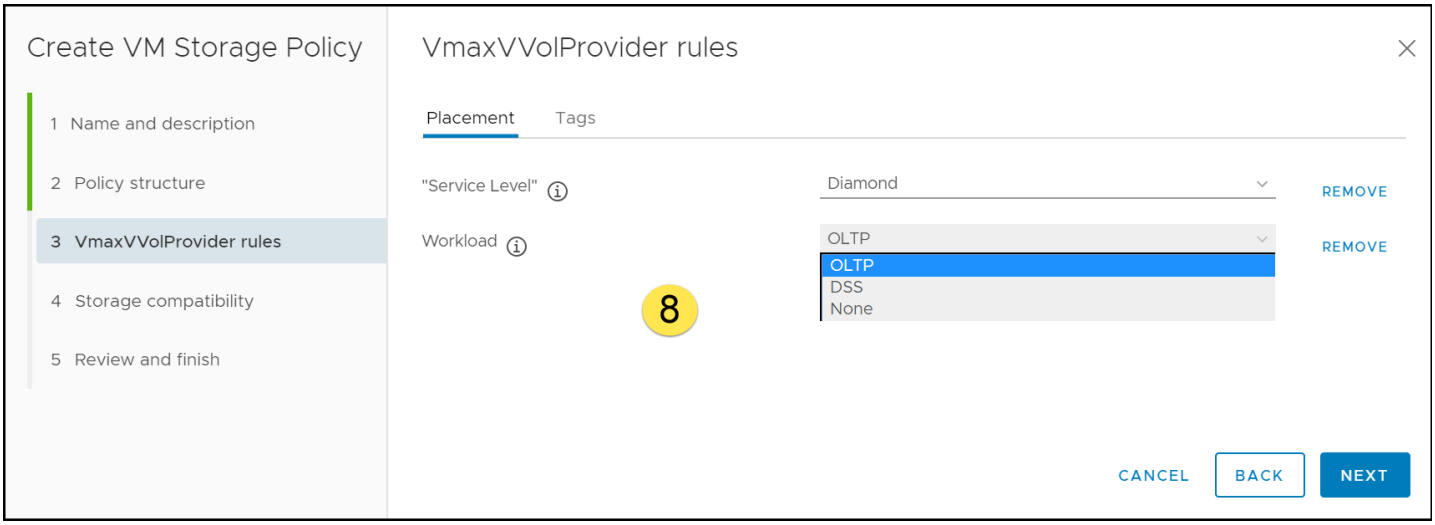

Figure 86. Creating a VM Storage Policy - step 8

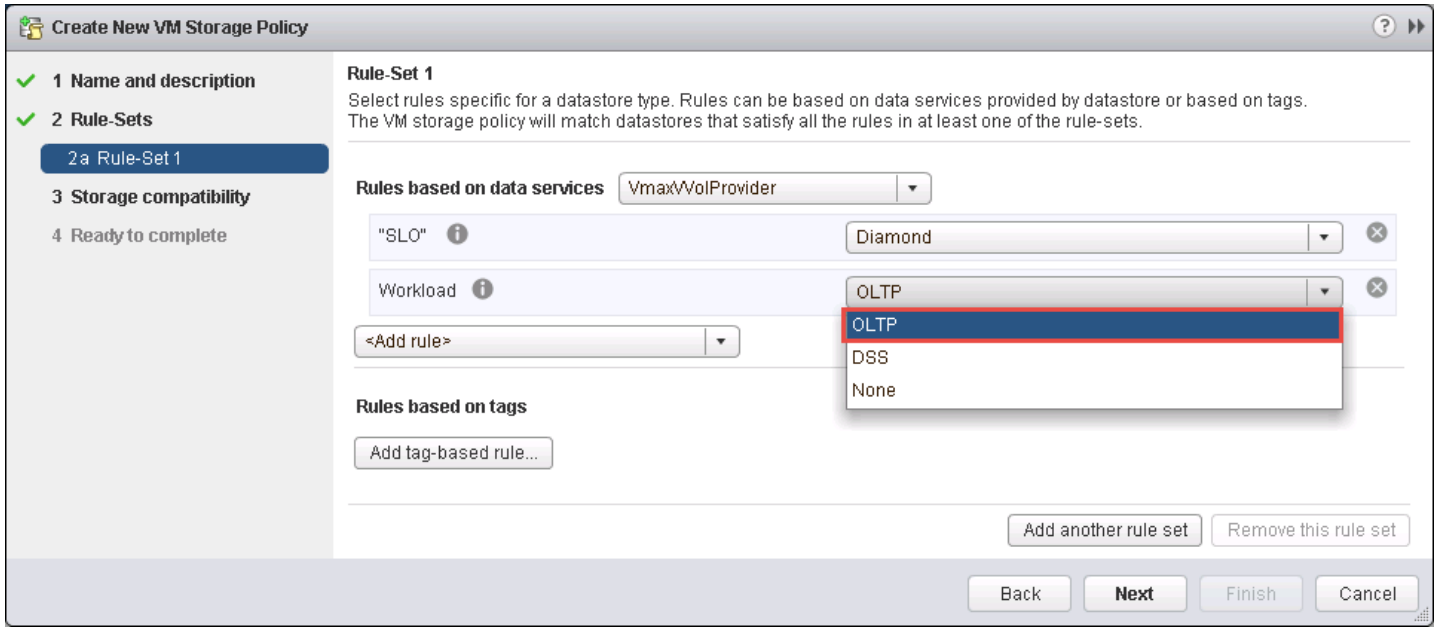

Figure 87. Creating a VM Storage Policy - step 8 (vSphere 6.5)

VM Storage Policy Step 9

VMware now takes the supplied parameters and compares it against the available vVol datastores to see if any are compatible. In Figure 88 the VVol_Finance_Datastore is compatible with the Diamond SL, while in Figure 89 the VVol_Demo_datastore is compatible with the Diamond+OLTP SLO.

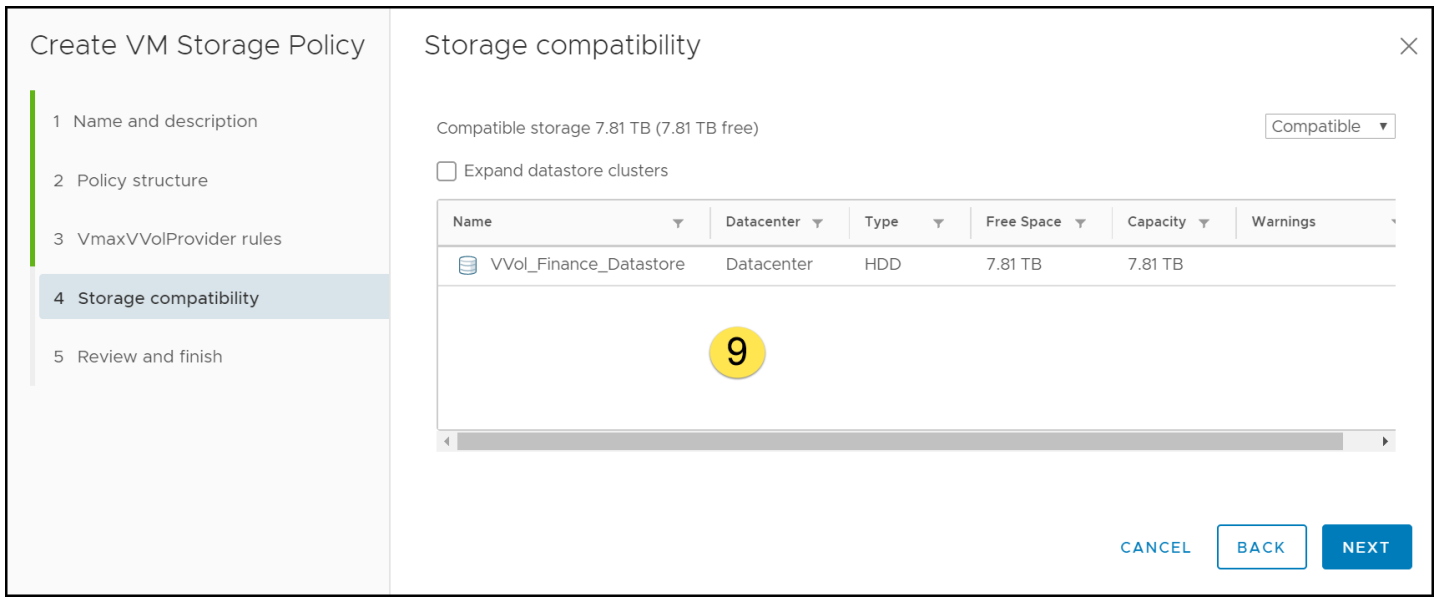

Figure 88. Creating a VM Storage Policy - step 9 (vSphere 6.7+)

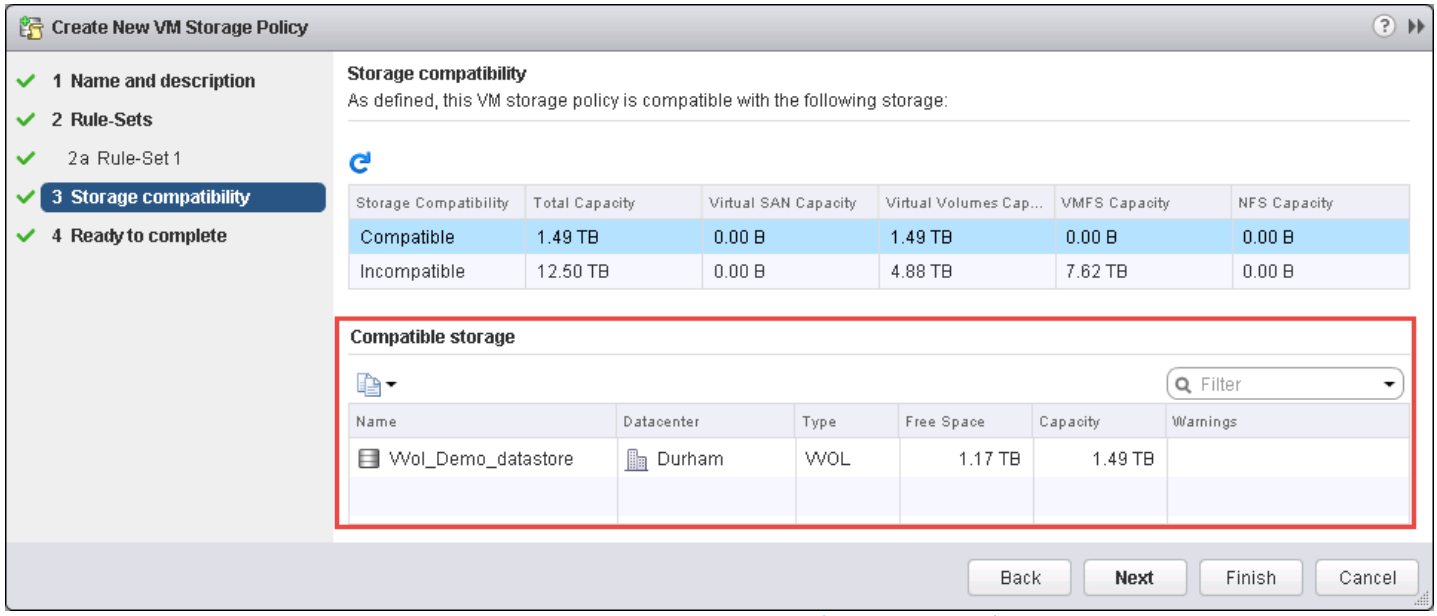

Figure 89. Creating a VM Storage Policy - step 9 (vSphere 6.5)

VM Storage Policy Step 10

A summary page in Figure 90 and Figure 91 complete the VM Storage Policy.

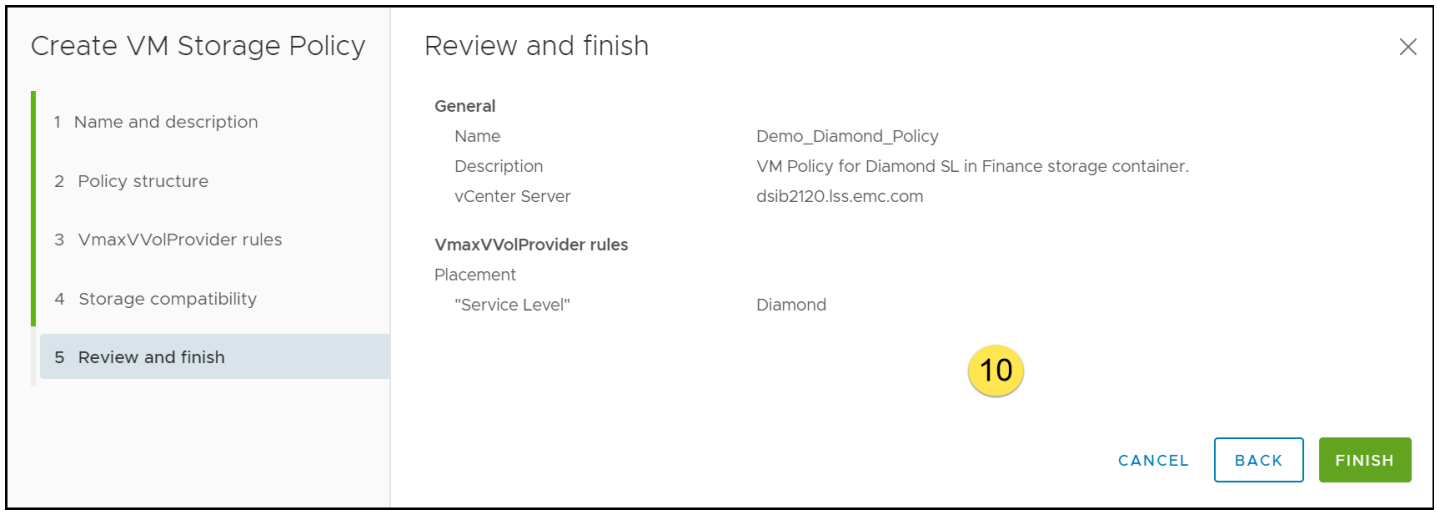

Figure 90. Creating a VM Storage Policy - step 10 (vSphere 6.7+)

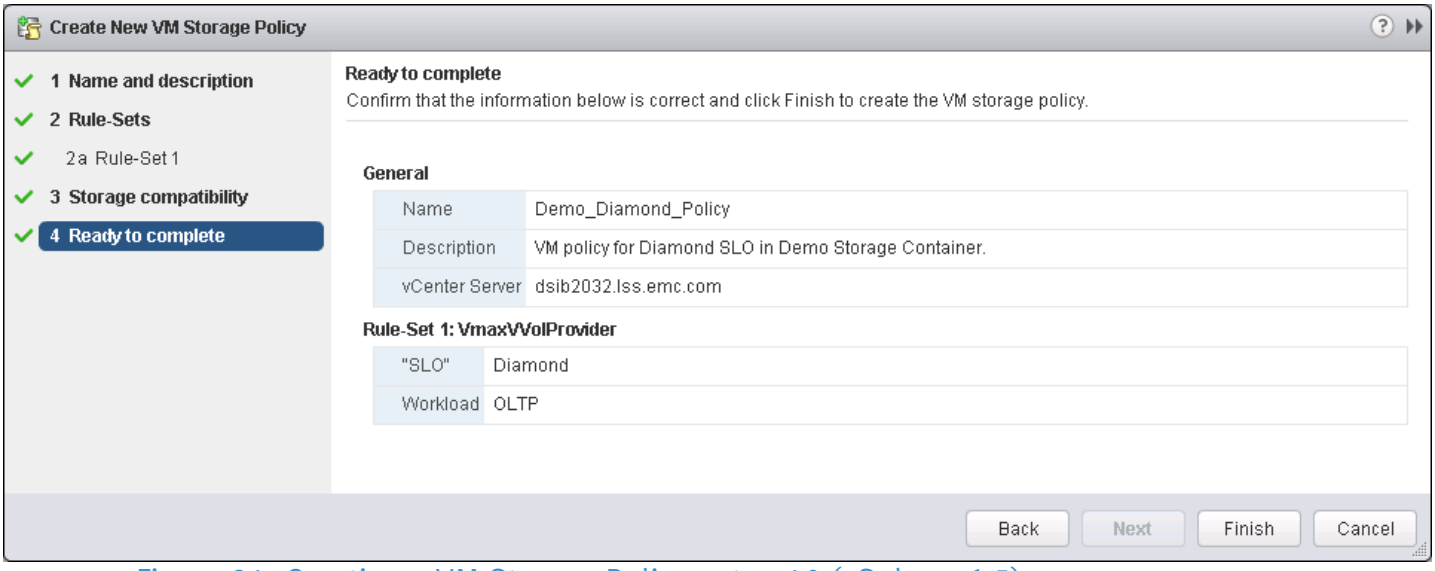

Figure 91. Creating a VM Storage Policy - step 10 (vSphere 6.5)

Creating a VM with vVol storage

Creating a VM using vVol storage in the vSphere Client is no different than creating a VM with VMFS storage. As both types of storage are represented by datastores, it is simply a matter of selecting which datastore type the user wishes to use for the VM. It is essential for vVols, however, to utilize the VM storage policy to not only select the correct vVol datastore, but ensure the correct SL is sent with the creation command. Rather than walk through the entire creation of the VM, only step 4 from the wizard is included below to illustrate assigning the proper SL.

The example below of creating a VM using vVol storage utilizes a single Hard disk (vmdk). If a VM has more than one vmdk, each vmdk could be assigned a different VM storage policy, and thus SL. For example, a user creating an application VM with 2 vmdks might select the VM storage policy for the Bronze SL for the OS vmdk, while assigning the Diamond SL to the application vmdk.

VM Creation

In step 4 of the VM wizard, the storage is selected for the VM. By selecting a VM storage policy as in Figure 92, the user can ensure that the vmdks that make up the VM will be assigned the desired SLO in the vVol datastore.

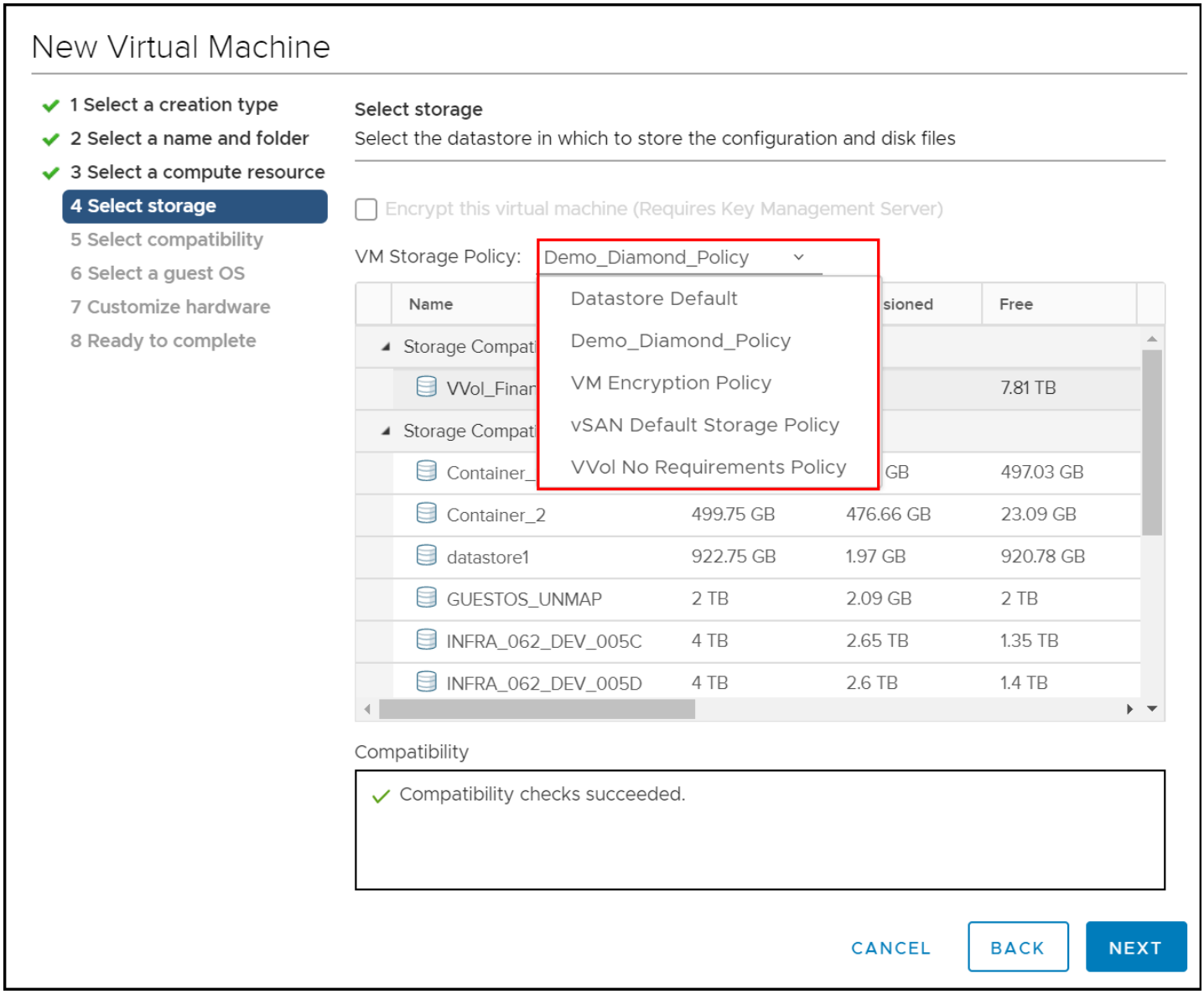

Figure 92. Creating VM – step 4 – Select storage

For instance, in the example in Figure 93, the VM storage policy "Demo_Diamond_Policy" is selected. This storage policy was previously created with the diamond SLO in Creating a VM Storage Policy for vVols. Once selected, VMware determines which vVol datastores are compatible with the policy. In this example, only one datastore, VVol_Finance_Datastore, shows as compatible. When the user selects that datastore with that VM storage policy, the vVol will be assigned the SLO of diamond.

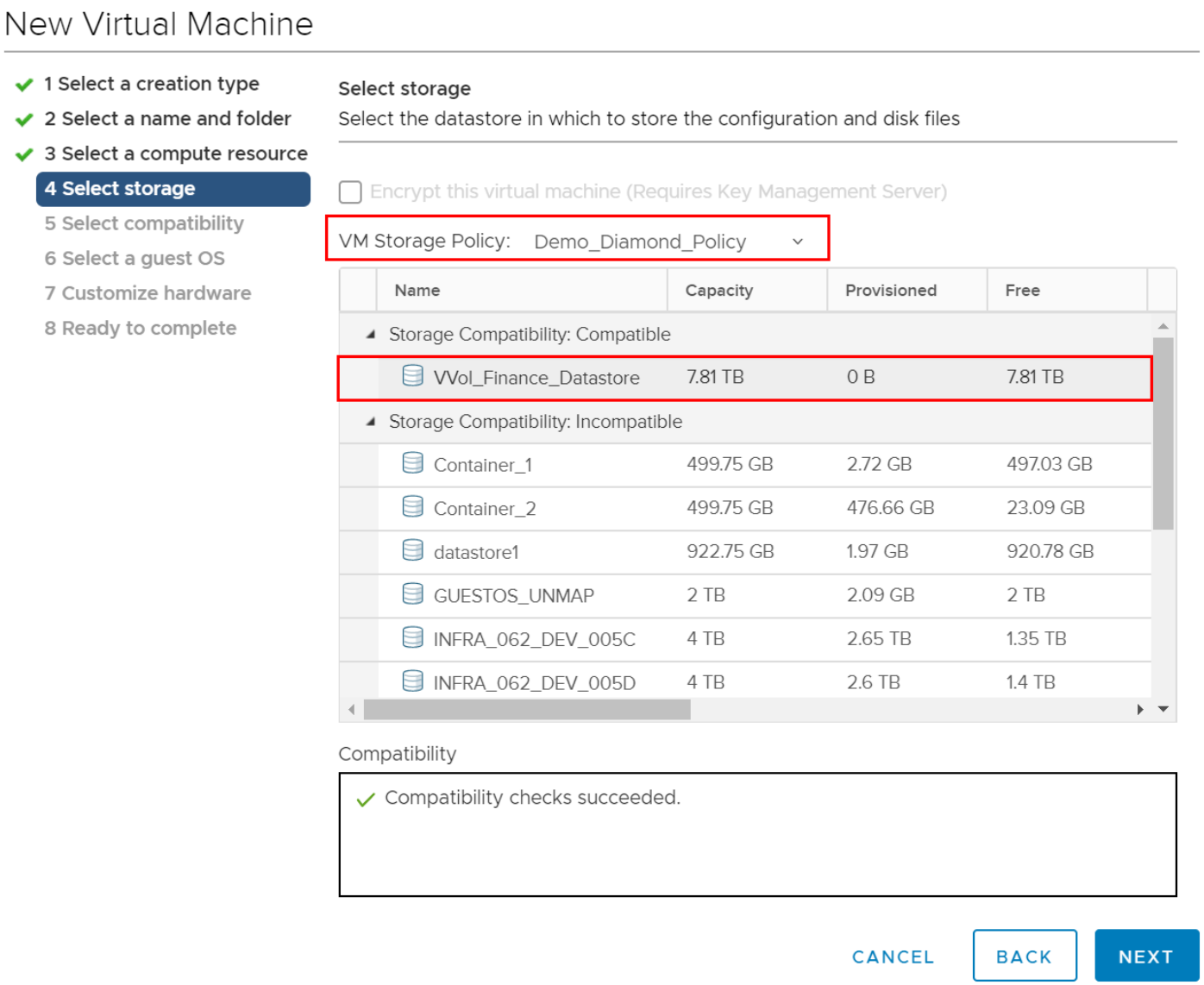

Figure 93. Creating VM – step 4 – VM Storage Policy

Upon creation, a VM that is created in a vVol datastore will be comprised of two initial vVols – a *data* vVol for the Hard disk which is the size chosen (16 GB in this case), and a 4 GB *config* vVol for the configuration files (e.g. vmx, vmsd). A third *swap* vVol will be generated when the VM is powered on. It is the size of the VM memory (1 GB in this case). Figure 94 displays the contents of the VM "VVol_Linux" in the vVol datastore VVol_Finance_Datastore. Note that despite the many files listed, there are only 3 vVols that are created on the VMAX array. The config vVol is comprised of multiple files, namely the VM metadata, which accounts for the other files in the directory.

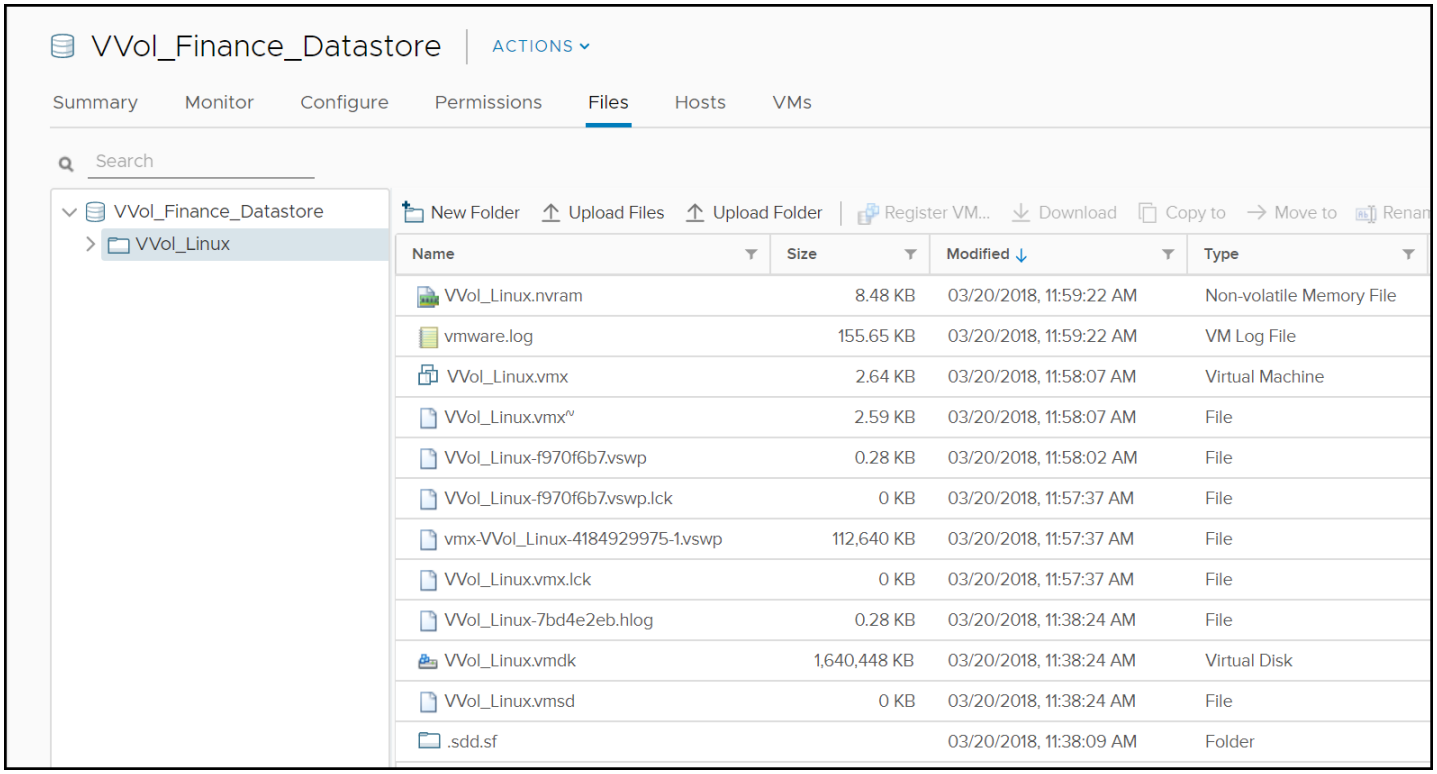

Figure 94. VM contents in a vVol datastore

VMware disk type

Another important thing to notice about the VVol_Linux.vmdk file in Figure 94 is that it is only 1.6 GB, rather than the actual size of the vmdk, 16 GB. This is because by default all vVols are created with a thin disk type. The vmdks will not grow unless data is actually written to them. In fact, even if the disk type is changed to “thick”, or an existing vmdk inflated, the size will not change.

In a traditional VMFS environment, the default disk type is zeroedthick (aka lazy), and any vmdk created with that type will show the created size in the datastore. In this case the vmdk would show 16 GB, even if no data was written to it. A zeroedthick disk reserves the space in the VMFS datastore, though it does not actually allocate the space on the array, an important distinction. By contrast, a thin disk type in VMFS does not reserve any space on the datastore and only grows as data is written to it. This can be dangerous in a VMFS environment because a datastore could fill before a thin vmdk is full (due to other vmdks in the datastore).

It might appear at first glance that thin vmdks work the same way in a vVol datastore as VMFS; however, they do not. In the VMAX implementation of vVols, a vVol datastore behaves differently. A thin vVol vmdk takes on the role of both disk types. A thin vVol vmdk reserves space in the vVol datastore and storage resource/storage container on the array. This ensures the VM will not run out of space as it is used. In the datastore, however, the thin vVol vmdk will only show growth as it is written to. The reason it works this way is because a vVol is not just a vmdk stored in a large datastore represented by a single device on the array. Every vVol is its own device

(TDEV). If that thin vVol vmdk was treated as truly thin, the backing device on the array would need constant resizing. That would be an incredibly expensive action in terms of performance. Instead the device is created with the requested size right away. It can be resized, of course, but the entire space of the vmdk is ready to be written to immediately. It is critical to remember, however, that storage resources do not reserve space in the SRP. So, while the space is reserved in the vVol datastore, if there is no available space in the SRP, the vVol will fail to allocate a new extent. In that case, of course, your entire array is out of space since most boxes have a single SRP.

Note that while vSphere 6.0 permits the use of eagerzeroedthick, Dell EMC recommends using the default of thin. In vSphere 6.5 the vmdk options have changed so the only options available are thick (zeroedthick) or thin (default).

Changing VM Storage Policy for VM

During the course of regular business operations, the performance requirements of an application may change. Before virtual volumes, if a change in SLO was needed it could only be done at the VMFS datastore level which impacted all VMs on that datastore. Alternatively, the VM could also be migrated off the source datastore to one with the required SLO, but that meant a potential decrease in performance of the VM during the move. With virtual volumes a single VM can now be assigned a new SLO without impacting any other VMs in the environment. In fact, a single vmdk of that VM could be assigned a new SLO. There is a very simple process to do this.

Changing the storage policy of a VM or VM disk within the same vVol datastore does not initiate a Storage vMotion. The array reassigns the vVols on the backend as required; however, if a VM is moved between vVol datastores, regardless if those datastores are on the same array, a Storage vMotion is required.

The following walks through changing the policy for a vmdk (Hard disk).

Change Storage Policy for VM Step 1

Access the VM from the left-hand menu. Select the Configure tab on the right and the Policies menu on the left. Each Hard disk is listed in this panel. In the right-hand corner start by selecting the EDIT VM STORAGE POLICIES button as in Figure 95.

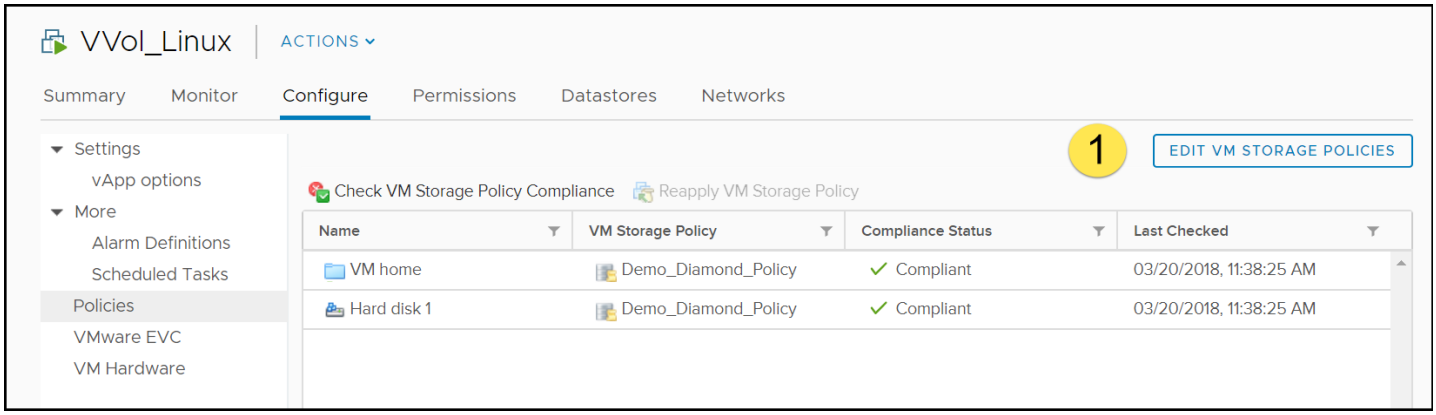

Figure 95. Changing a Storage Policy for VM - step 1

Change Storage Policy for VM Step 2-4

When changing the storage policy, there are two options available. By default, changing the VM storage policy will apply to all disks. Simply use the drop-down box highlighted in the red box in step 2 in Figure 96, select the new policy, and select OK. The second option is available in step 3 by toggling the option in the right-hand corner, Configure per disk. Enabling this will show the drop-down box for each disk in the VM so that individual Hard disks can be changed. In this example in step 4 the Hard disk 1 is being changed from Demo_Diamond_Policy to Demo_Gold_Policy. Select OK to apply.

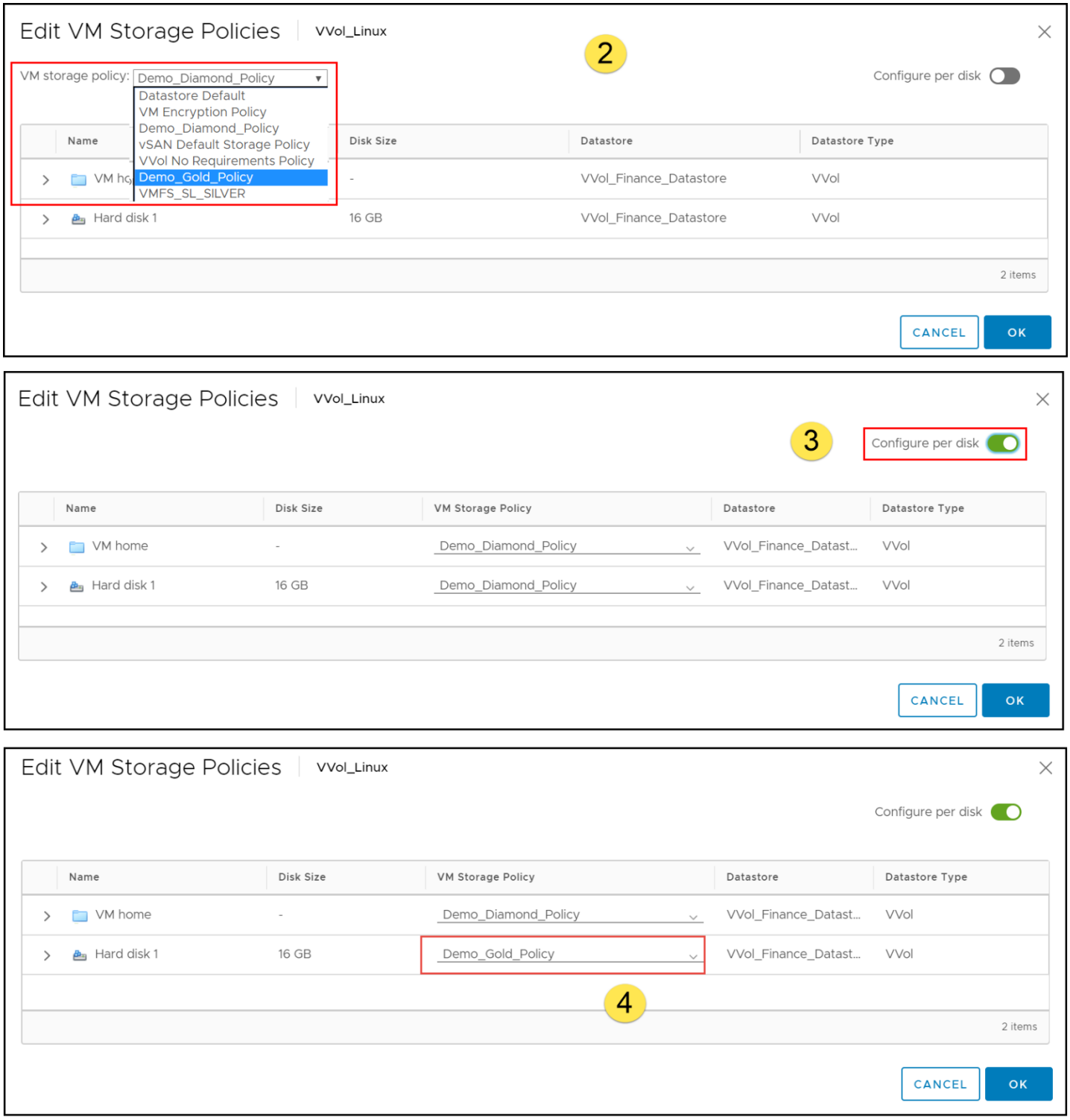

Figure 96. Changing a Storage Policy for VM - step 2-4

VMware makes the appropriate calls to the VASA Provider which calls to the array to move the vVol to the appropriate SLO, only changing the Hard disk 1 location but

leaving the configuration files in the diamond SLO. The two storage policies for each vVol are seen in Figure 97.

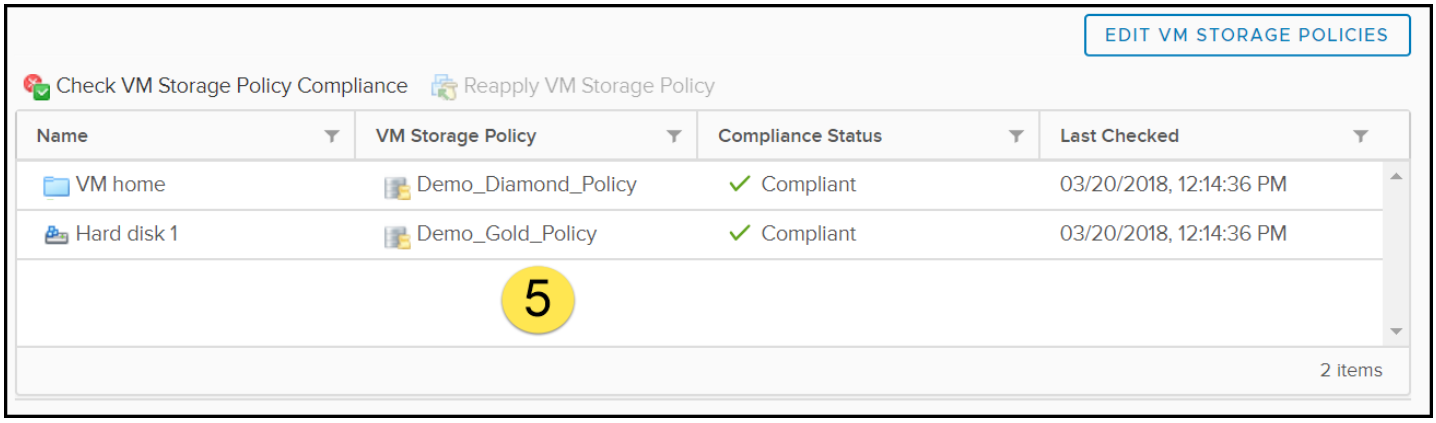

Figure 97. Changing a Storage Policy for VM - step 5

VMware CLI for vVols

VMware offers some ability to view the setup of the VMAX vVols on a host. The command, `esxcli storage vvol`, pictured in Figure 98, has five available namespace commands.

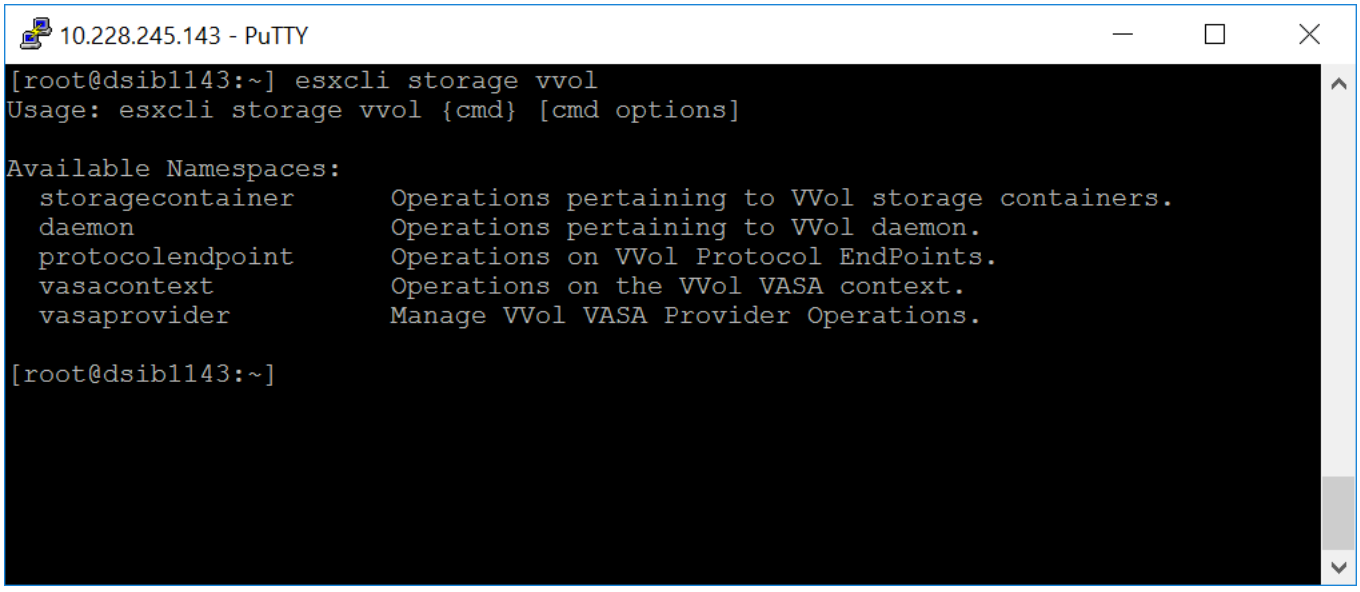

Figure 98. VMware esxcli options for displaying vVol information

Most of these commands are simply to list objects such as containers, protocol endpoints, or even the VASA Provider; however, the *vasacontext* command gets the vCenter UUID, while the *daemon* command can cause disruption as it unbinds all virtual volumes from the known VASA Provider. The three most useful commands and their output are shown in Figure 99.

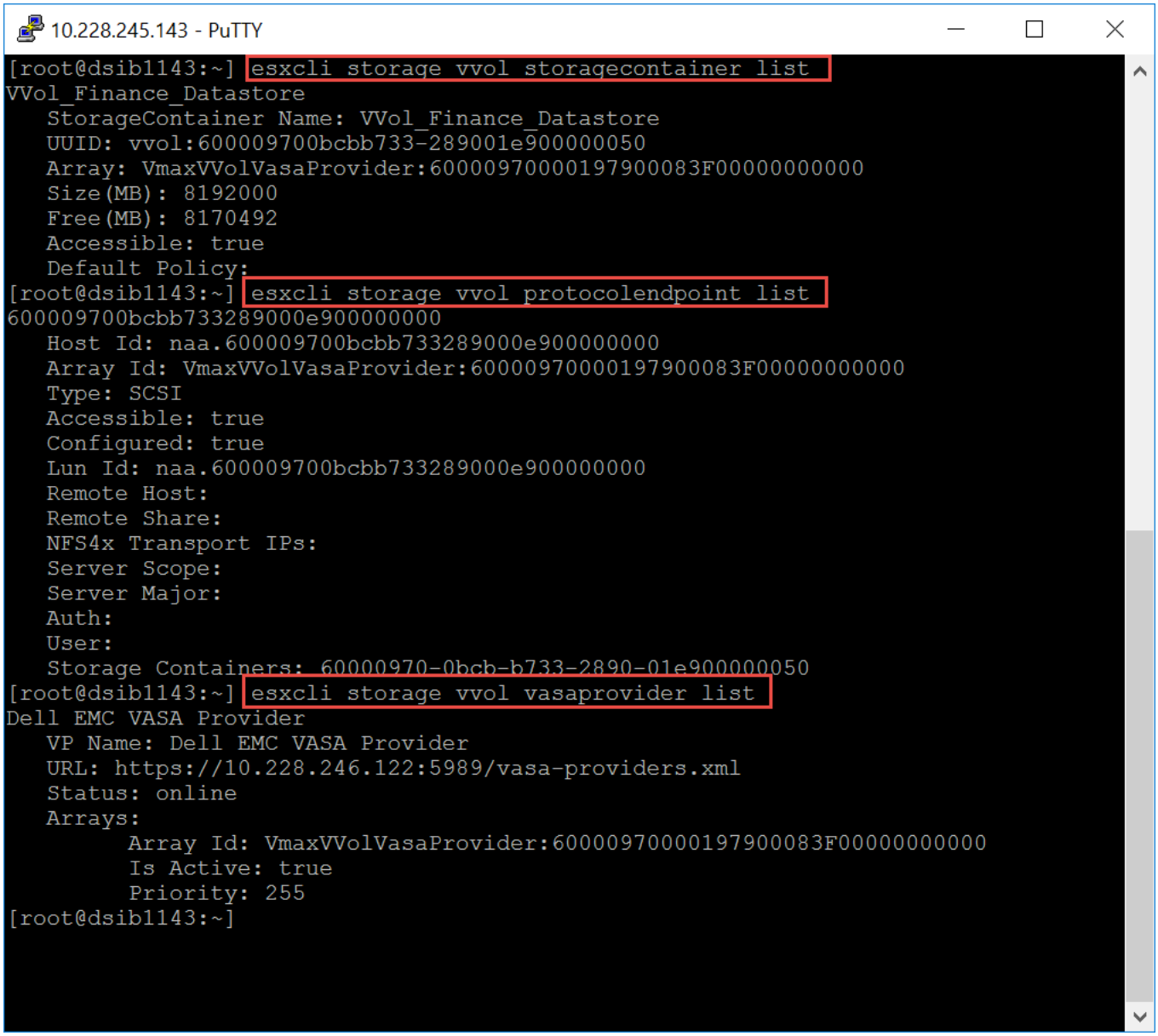

Figure 99. Namespace commands for vVol object listing

vVol datastores in a cluster

In a traditional VMFS environment, devices are presented in a single storage group to a VMware cluster. Each host in the cluster sees the same device and therefore when a datastore is created on that device, upon rescan each host will recognize the new datastore. With vVols, there is no storage group. Each host in a cluster is presented a unique PE to which vVols are bound. When creating a datastore, therefore, the wizard behaves similarly to NFS, and will present the hosts in the cluster as available for mounting the datastore. One simply checks the boxes of the hosts which have a PE as in Figure 100.

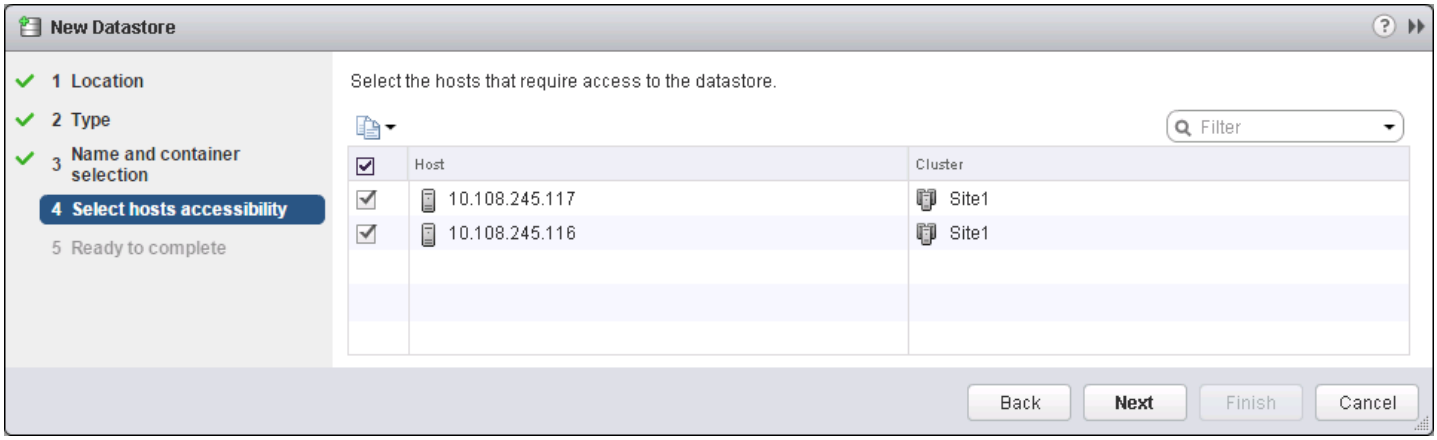

Figure 100. Creating vVol datastores in a cluster

Note that VMware does not validate that each host has a presented PE, and therefore if an attempt is made to mount the datastore to a host without a PE, it will show as inaccessible (Figure 101).

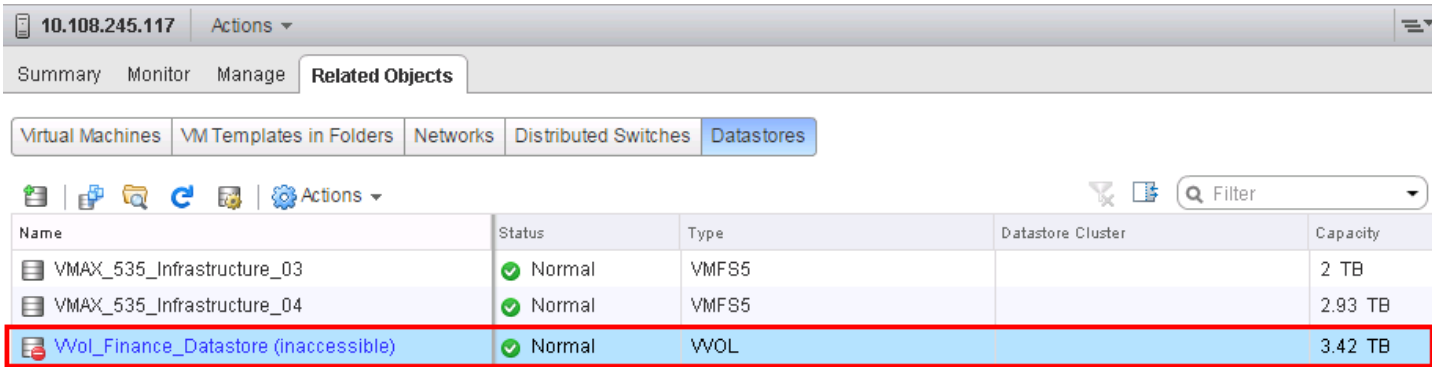

Figure 101. ESXi host with no presented PE

VMware High Availability (HA)

VMware HA is supported with vVols, though there are some important things to keep in mind. Firstly, as has been made clear, each host must see a unique PE. ESXi hosts in any cluster, including HA, may not share a PE. Secondly, VM Component Protection (VMCP) is not supported with vVols and hence if APD or PDL situations arise, the way the host reacts may not be the same as it will be with VMFS. This includes, but not limited to, vVol VMs not failing over when an APD event is experienced.

Default profile and default Storage Policy for vVol datastores

There are two different default capabilities a vVol datastore may have – a default profile and a default Storage Policy. Setting these defaults is covered in the next two sections.

Default profile

When a vVol datastore is created, a default profile is automatically assigned to it which is taken from the available Capability sets. The default profile is used if a VM or

disk is created in the vVol datastore without specifying a VM Storage Policy. The capability sets can be found in the datastore detail under the tab Manage and sub-tab Settings/Capability sets. In this example in Figure 102, there are two available capability sets from which the default profile can be drawn.

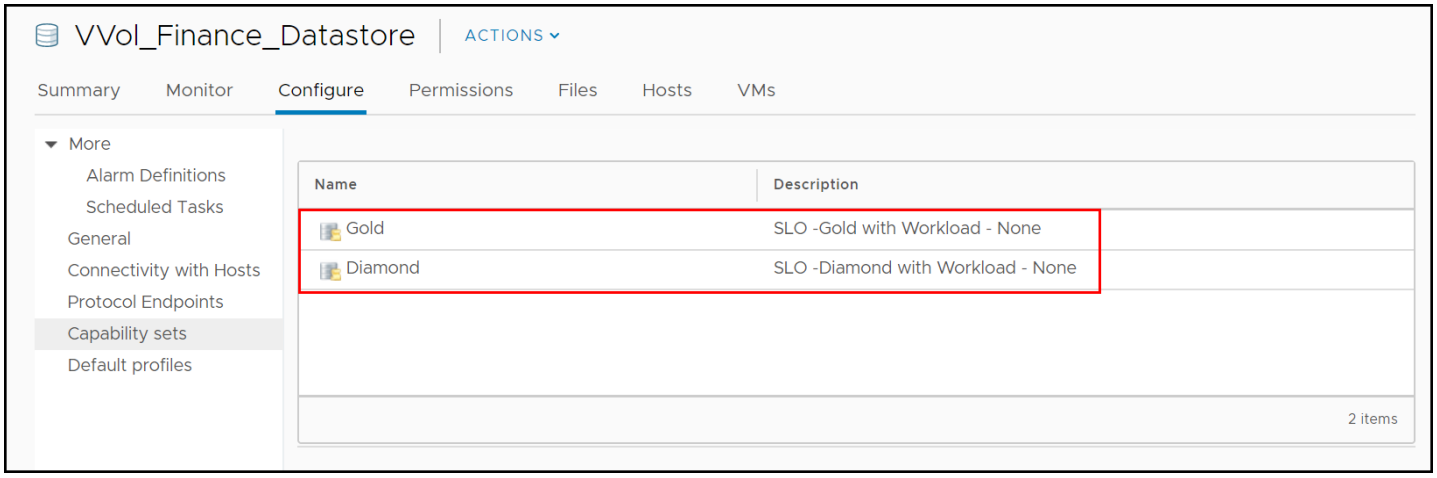

Figure 102. vVol Capability sets

Dell EMC will always use the least performant capability as the default profile. In this case, therefore, that is Gold. For the VVol_Finance_datatstore this is shown in Figure 103.

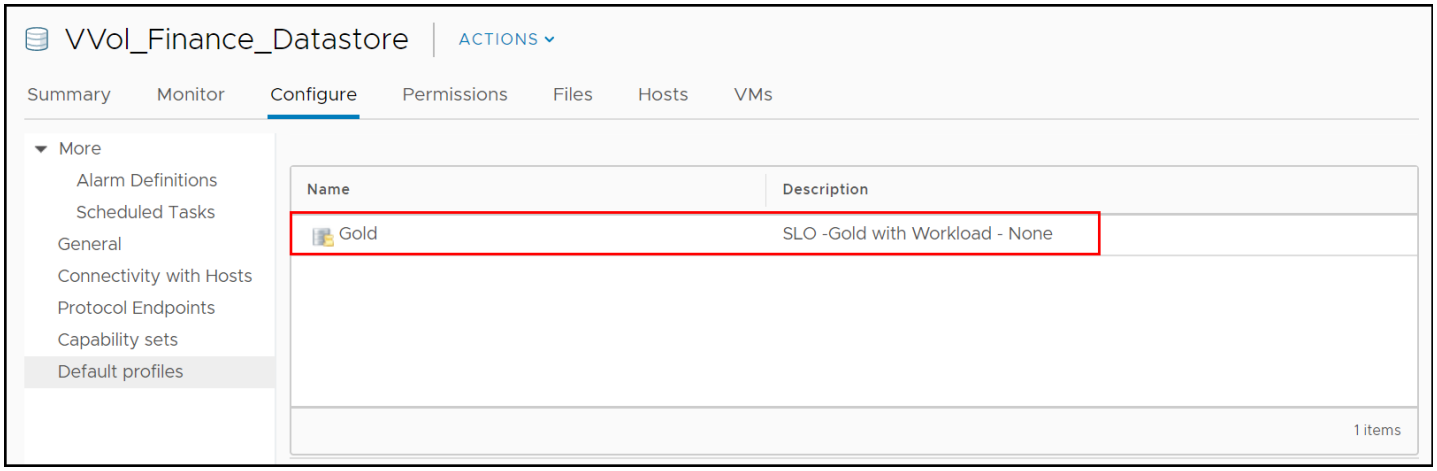

Figure 103. Default profile for vVol datastore

The default profile cannot be changed unless a storage resource is added to the storage container that is less performant than the current profile. For example, when the Optimized SLO is added to the storage container for the VVol_Finance_datastore, the profile is updated as in Figure 104.

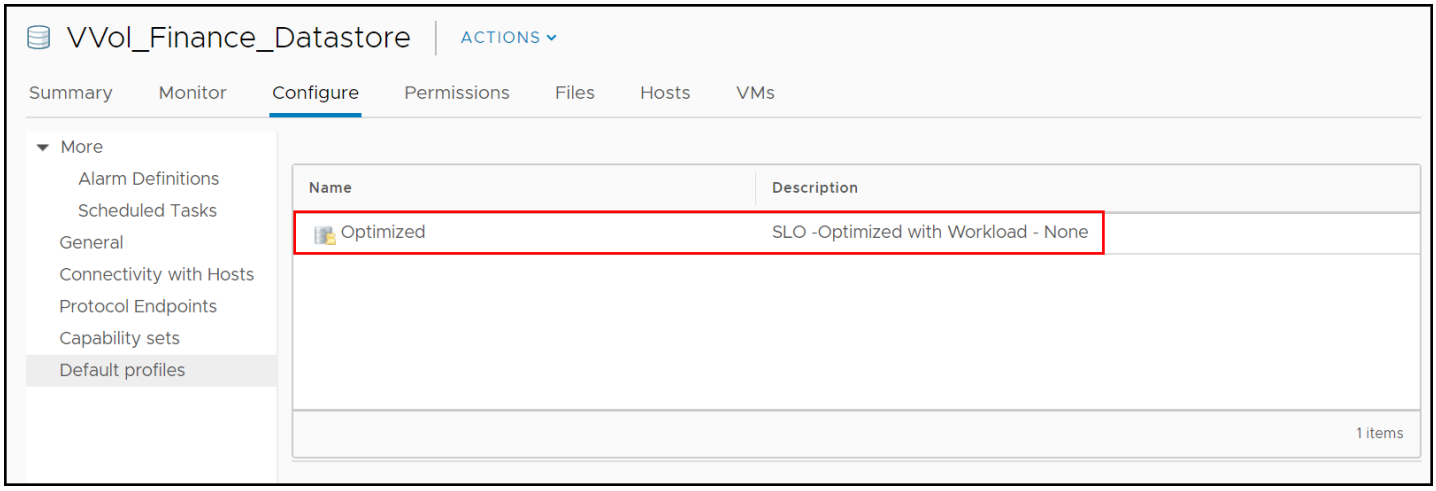

Figure 104. Changing the default profile

Default Storage Policy

In order to override the default profile, a storage policy can be set which will take precedence over the default profile. The initial policy for a vVol datastore is "vVol No Requirements Policy" present in Figure 105. As set, this policy will mean the default profile will be used.

As the Optimized SL has no response time target, it is considered to be the lowest SL if it is available in the storage container; however, this does not hold true for the swap file. Unless Optimized is the only SL available, the next lowest SL will be used for swap.

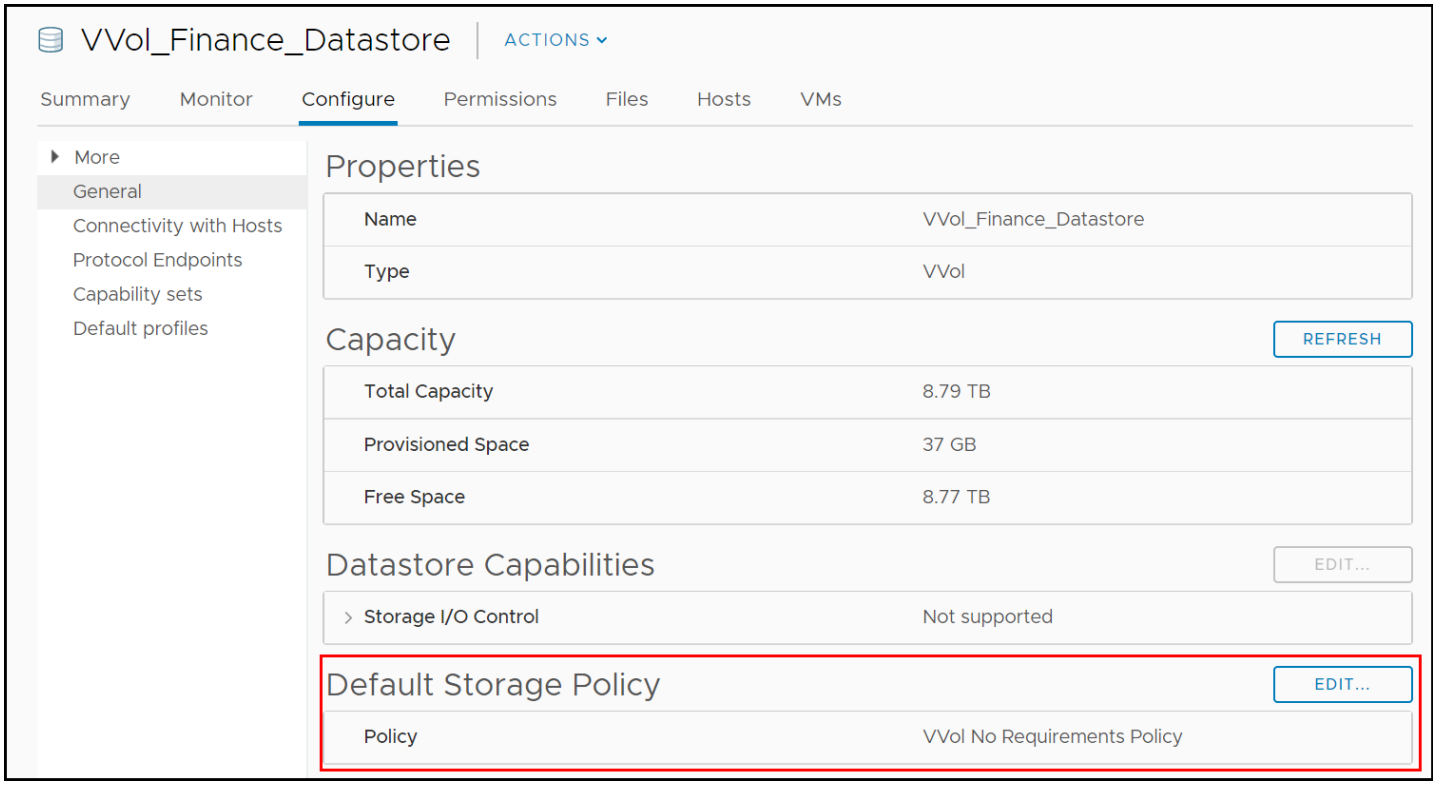

Figure 105. Default Storage Policy

To change the policy to one of the storage policies previously created, start by selecting the EDIT button in Figure 105.

In the dialog box that appears in Figure 106, select the desired default storage policy from the available policies and select OK.

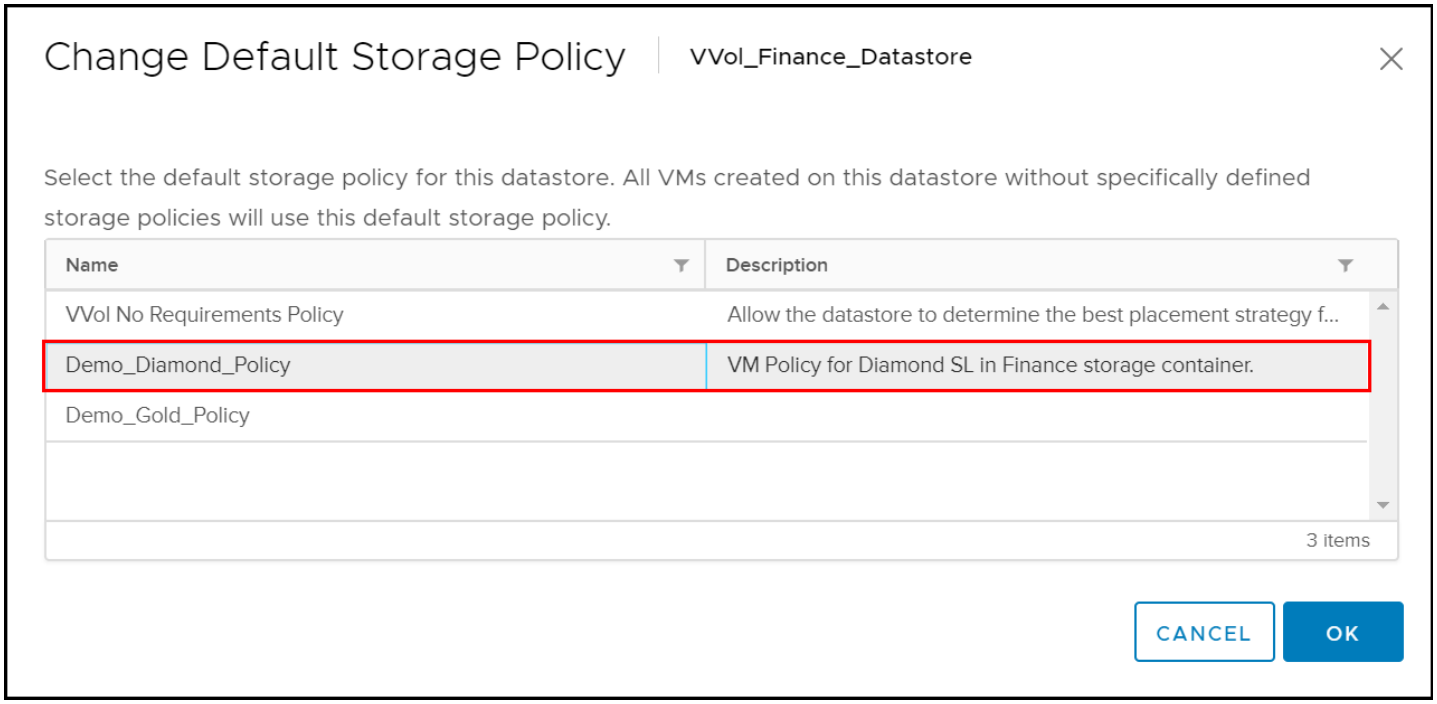

Figure 106. Select new default storage policy

The new policy is now set and can be seen in Figure 107. This policy will now be used when no policy is selected during VM or vmdk creation.

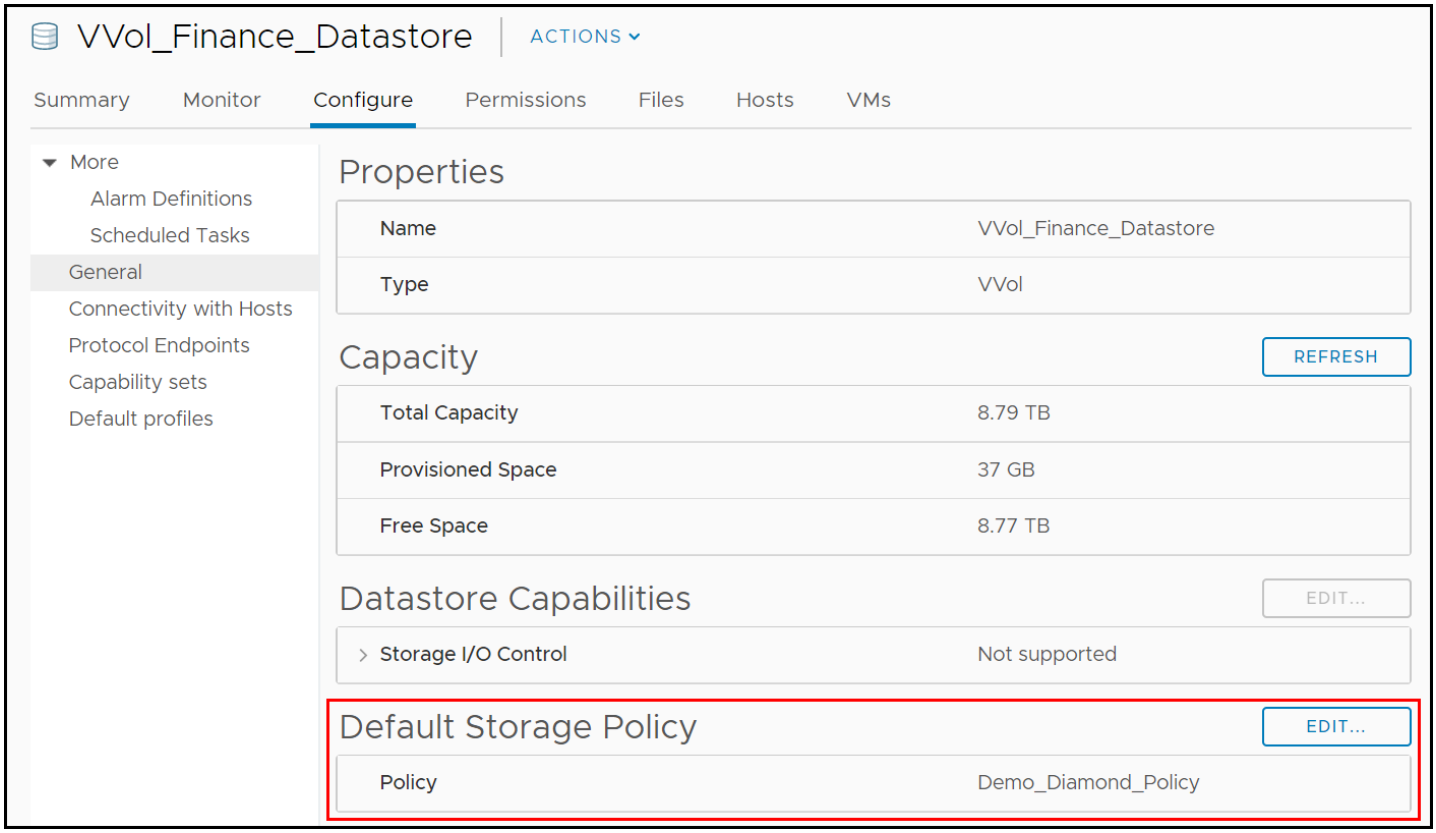

Figure 107. New Default Storage Policy

VMFS and vVol Cloning/Migrations

Cloning and migration of VMs between vVol and VMFS datastores is fully supported through the normal Storage vMotion function. The following table, Table 2, contains the most common tasks related to cloning and migrating virtual machines and what APIs (simplified to VAAI or VASA) are used to accomplish those tasks. For tasks that can use VAAI (XCOPY), host-based copy (software) will be used on the VMAX and PowerMax since XCOPY is not supported.

Table 2. Cloning/Migration tasks and functions

Task	Default Function	Failback Function
Clone from VMFS to vVol	VAAI	host-based copy
Clone vVol to VMFS	host-based copy	
Migrate vVol to VMFS	host-based copy	
Clone vVol to vVol in same container	VASA	host-based copy
Clone vVol to vVol in different container, same array	VASA	VAAI, host-based copy
SvMotion (powered on) without snapshots	VAAI	host-based copy
SvMotion (powered on) with snapshots	VASA, VAAI	host-based copy
SvMotion (powered off) without snapshots	VASA	host-based copy

SvMotion (powered off) with snapshots	VASA	host-based copy
Clone vVol to vVol in different container, different array	host-based copy	
Migrate vVol within the different container, different array	host-based copy	

Creating a VM Storage Policy for VMFS

The following provides a step-by-step process for creating a VM storage policy for VMFS datastores using VASA 1.0 capabilities of the VASA 2.0 Provider. This step-by-step will only use the vSphere 6.7 wizard. For differences between this version and vSphere 6.5, see the section "Creating a VM Storage Policy for vVols."

VM Storage Policy Step 1

Start by accessing the VM Storage Policies icon in the Home page of the vSphere Web Client as shown in Figure 108.

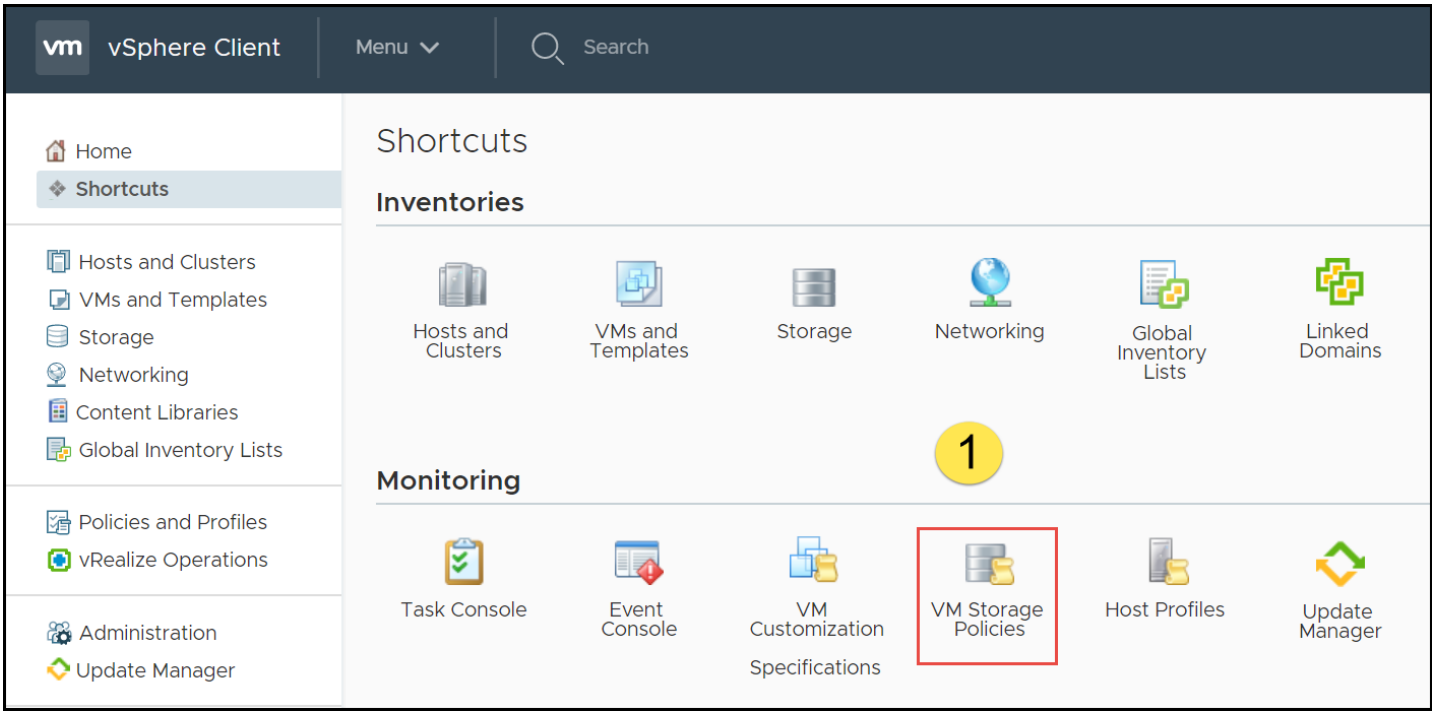

Figure 108. Creating a VM Storage Policy - step 1

VM Storage Policy Step 2

Next in step 2 in Figure 109 select the icon Create VM Storage Policy to create a new VM storage policy.

Figure 109. Creating a VM Storage Policy - step 2

VM Storage Policy Step 3

Step 3 formally starts the wizard. If the environment shares a Platform Service Controller then begin by selecting the appropriate vCenter. Enter a name for the policy, preferably one that reflects the capabilities that will be associated with the policy as this is the name the VM user will see. Finally, if desired enter a description. An example is shown in Figure 110.

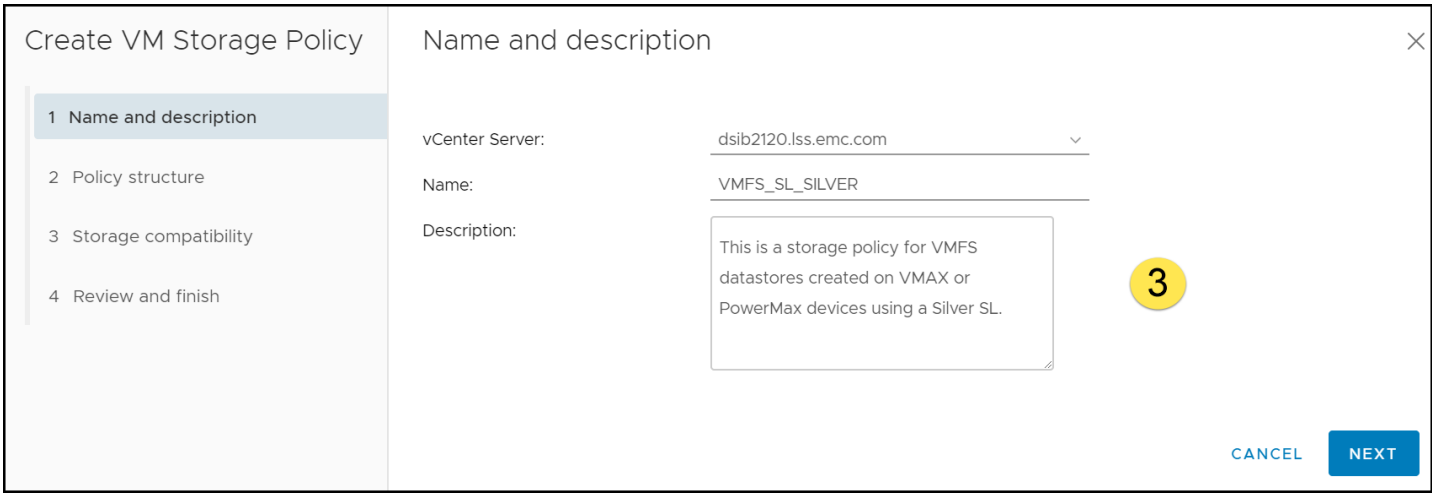

Figure 110. Creating a VM Storage Policy - step 3

VM Storage Policy Step 4

The Dell EMC VASA Provider advertises two different data services in step 4:

- VmaxvVolProvider
- VmaxvVolVasaProvider.VASA10

The difference between the two services is straightforward – the first supports VASA 2.0 and vVol functionality while the second supports the older VASA 1 capabilities. For this task, the VmaxvVolVasaProvider.VASA10 is used. The VmaxvVolProvider is covered in section: Creating a VM Storage Policy for vVols. Therefore, check the box for the VmaxvVolVasaProvider.VASA10 as demonstrated in Figure 111.

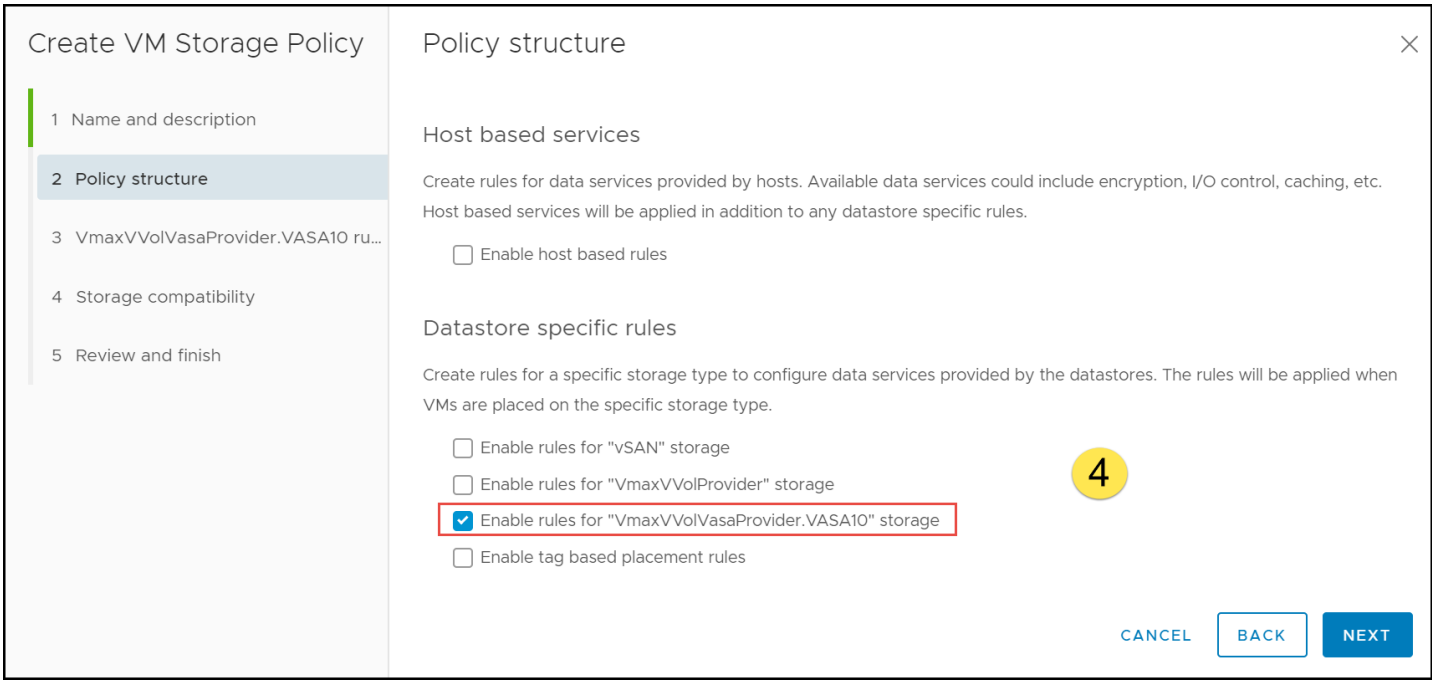

Figure 111. Creating a VM Storage Policy - step 4

VM Storage Policy Step 5

Once the data service is selected, the advertised capabilities may be added as rules for the policy. There is only a single rule for VASA 1 capabilities and that is SystemLabel.label. It will show all the SLOs available on the box in a drop-down list. Select the desired SLO for the storage policy, in this case Silver, as shown in Figure 112.

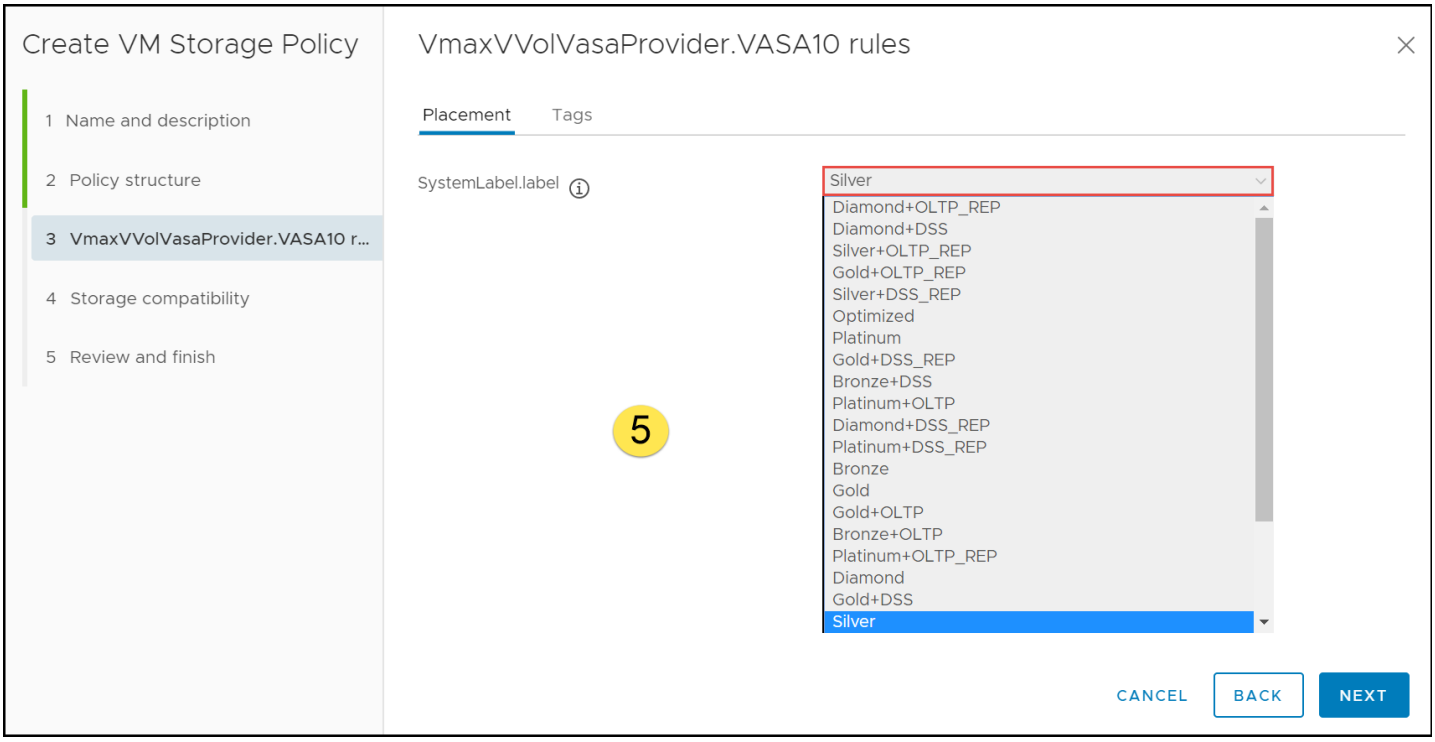

Figure 112. Creating a VM Storage Policy - step 5

VM Storage Policy Step 6

VMware now takes the supplied parameters and compares it against the available VMFS datastores to see if any are compatible. In Figure 113 the Silver_SL_Datastore is compatible with the Silver SLO.

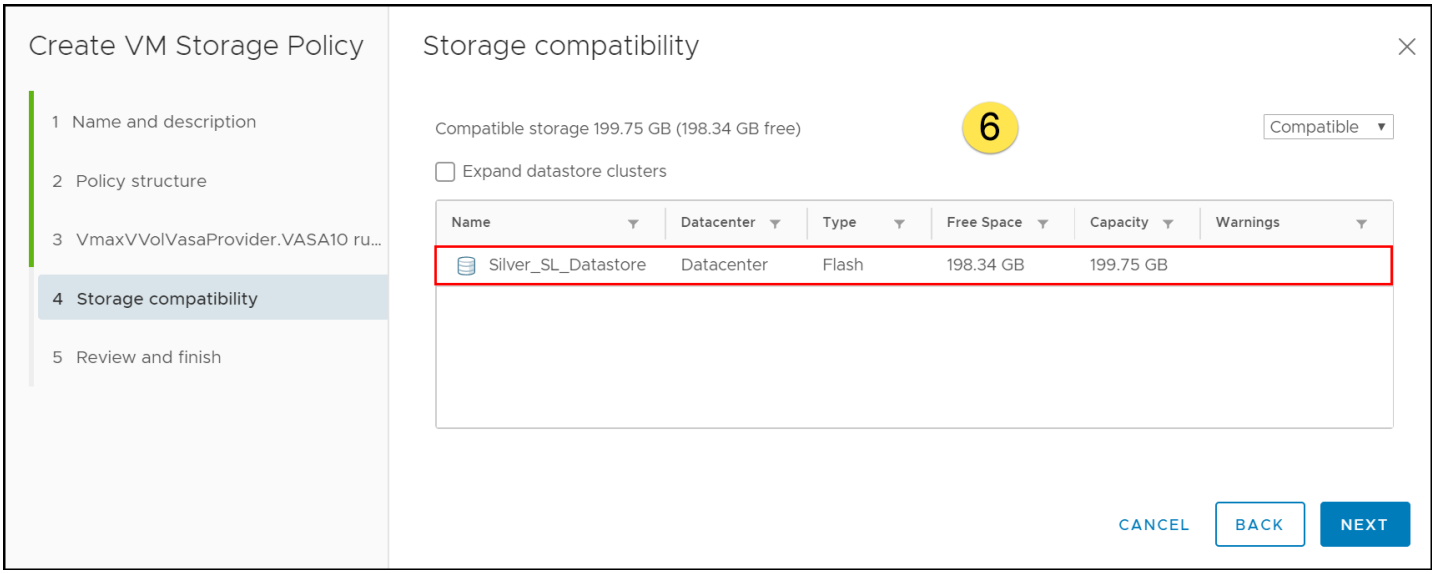

Figure 113. Creating a VM Storage Policy - step 6

VM Storage Policy Step 7

A summary page in Figure 114 completes the VM Storage Policy.

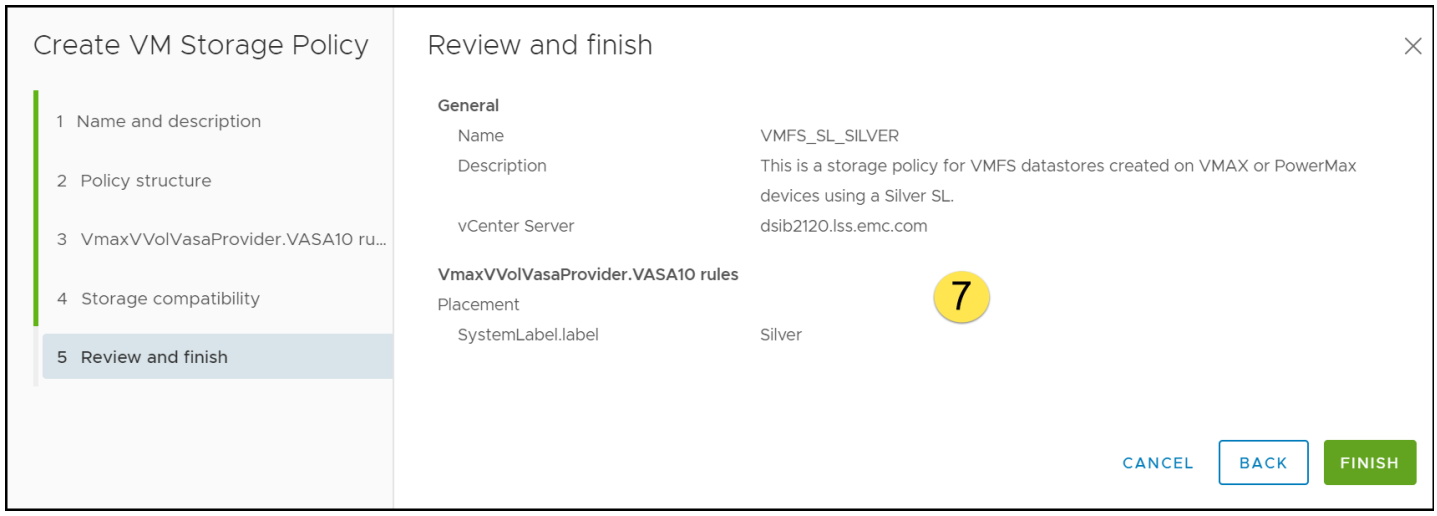

Figure 114. Creating a VM Storage Policy - step 7

Using VM Storage Policy in VM Creation

When creating, cloning, or migrating virtual machines, storage policies can be used to ensure proper placement of the virtual machines' virtual disk(s). Figure 115 shows the storage policy screen of the Create New Virtual Machine Wizard. This screen is very similar to the virtual machine migration or cloning wizard. A user can select a policy in the drop-down menu and the wizard will automatically sort the datastores according to their compatibility status with the selected storage policy.

If datastore clusters are configured they will be listed in place of their individual datastores. It is important to note that datastore clusters will only be assigned a storage capability by VASA if all included datastores are of the same type. If the cluster includes mixed-type datastores the cluster will be marked as incompatible. For this reason, it is highly recommended to only group datastores into clusters if they all have the same capabilities.

Furthermore, if Storage DRS is enabled on a datastore group the user will not have to specify which datastore in the cluster should be used. Storage DRS will offer recommendations at the end of the wizard as to which datastore would be preferred. These recommendations can be accepted or overridden.

Incompatible datastores or datastores may be chosen if desired—this is not prevented by vCenter.

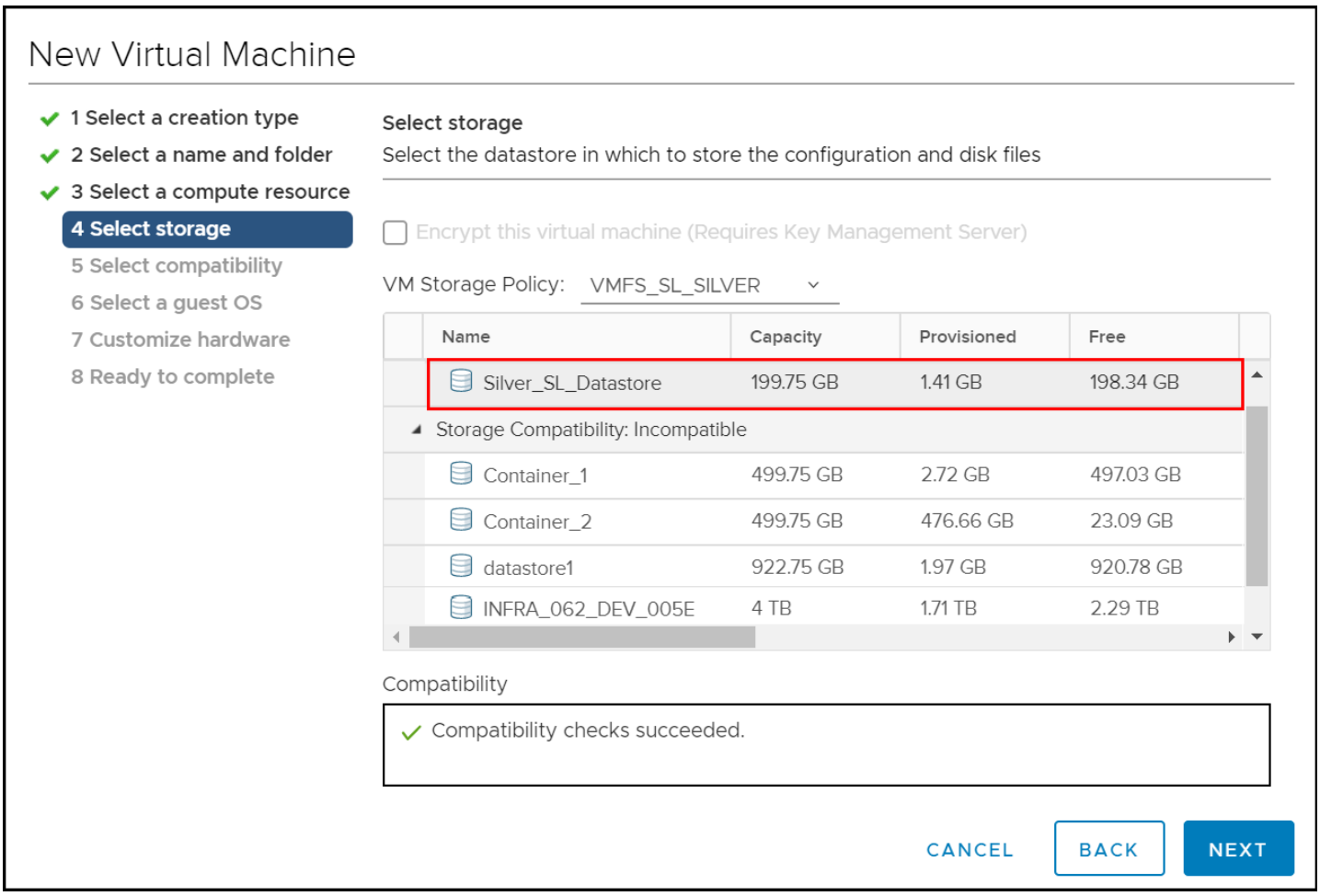

Figure 115. Placing a virtual machine according to a storage policy

Once a compatible datastore or datastore cluster has been chosen, the policy will be applied to all of the virtual machine's virtual disks. If a user wishes to remove the profile association or edit which profile is associated to a virtual machine this can be achieved by choosing to edit the setting of the virtual machine.

Figure 116 displays the properties of a virtual machine. The storage policy configuration can be edited from the Virtual Hardware tab. If the storage policy for a particular disk needs to be changed or assigned to a new disk, first use the drop-down box to select the policy.

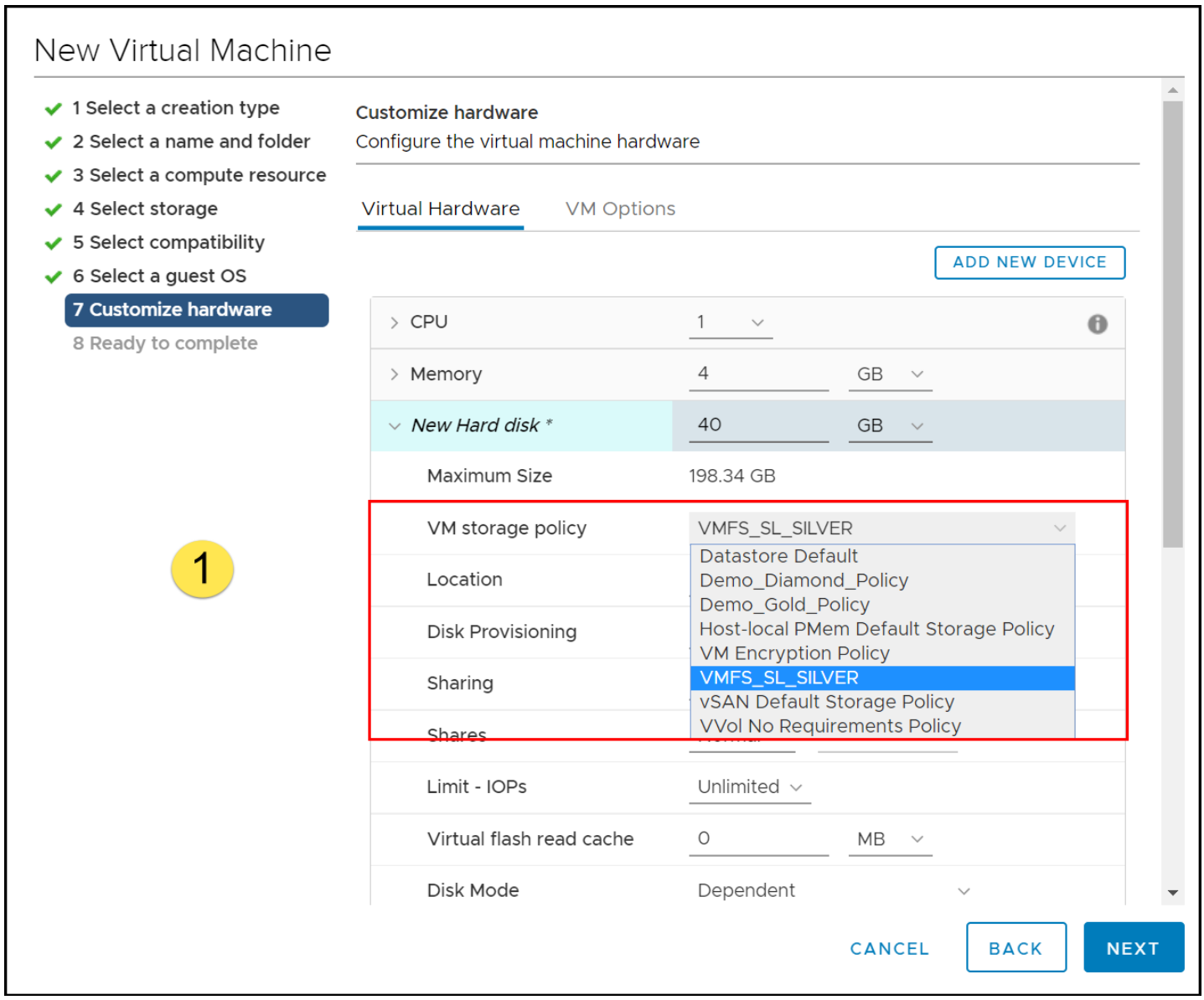

Figure 116. Assigning a new VM storage policy to a virtual machine – step 1

Next, under the Location row, select Browse. This is seen in Figure 117.

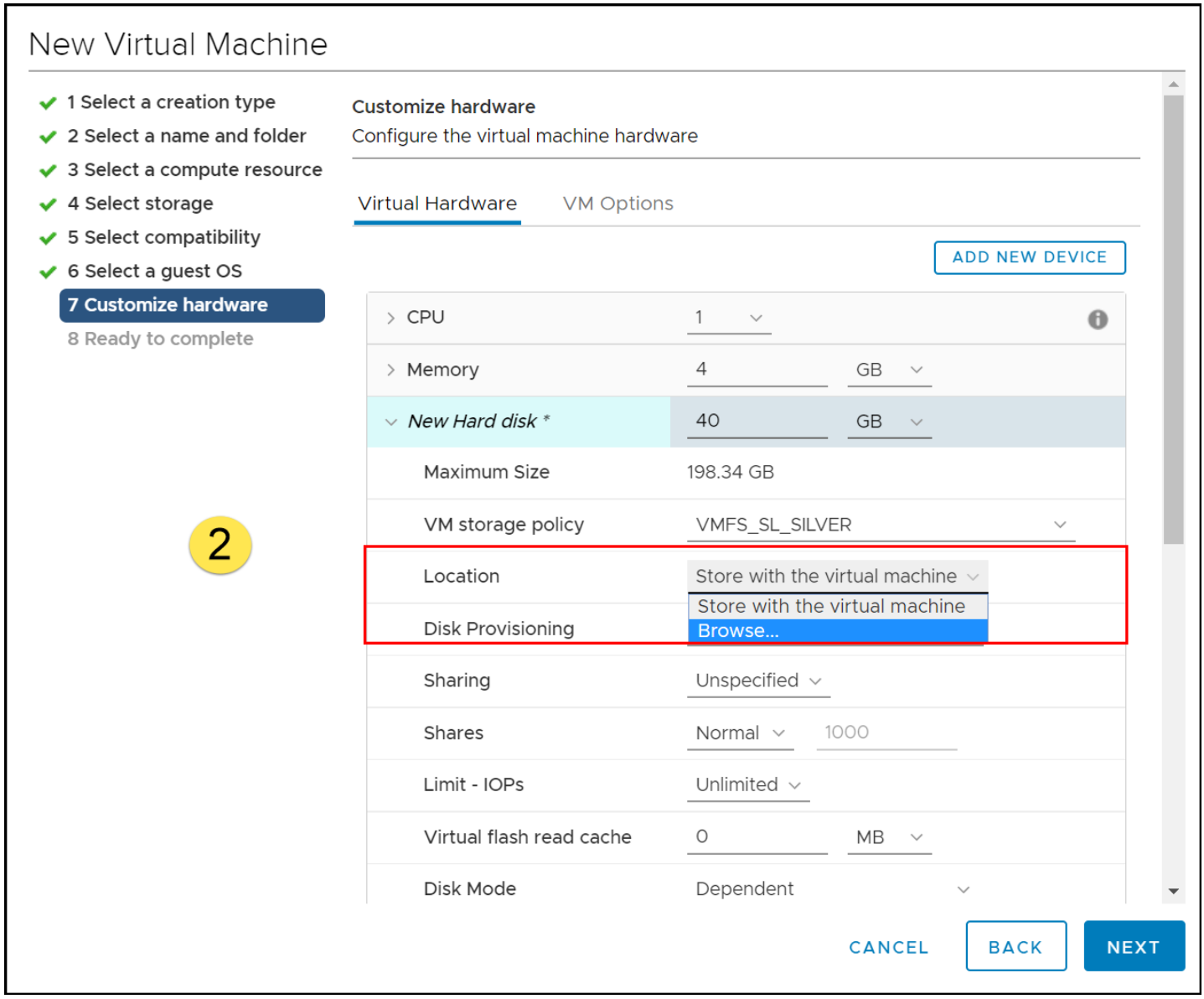

Figure 117. Assigning a new VM storage policy to a virtual machine – step 2

Finally select the new VM storage policy in Figure 118.

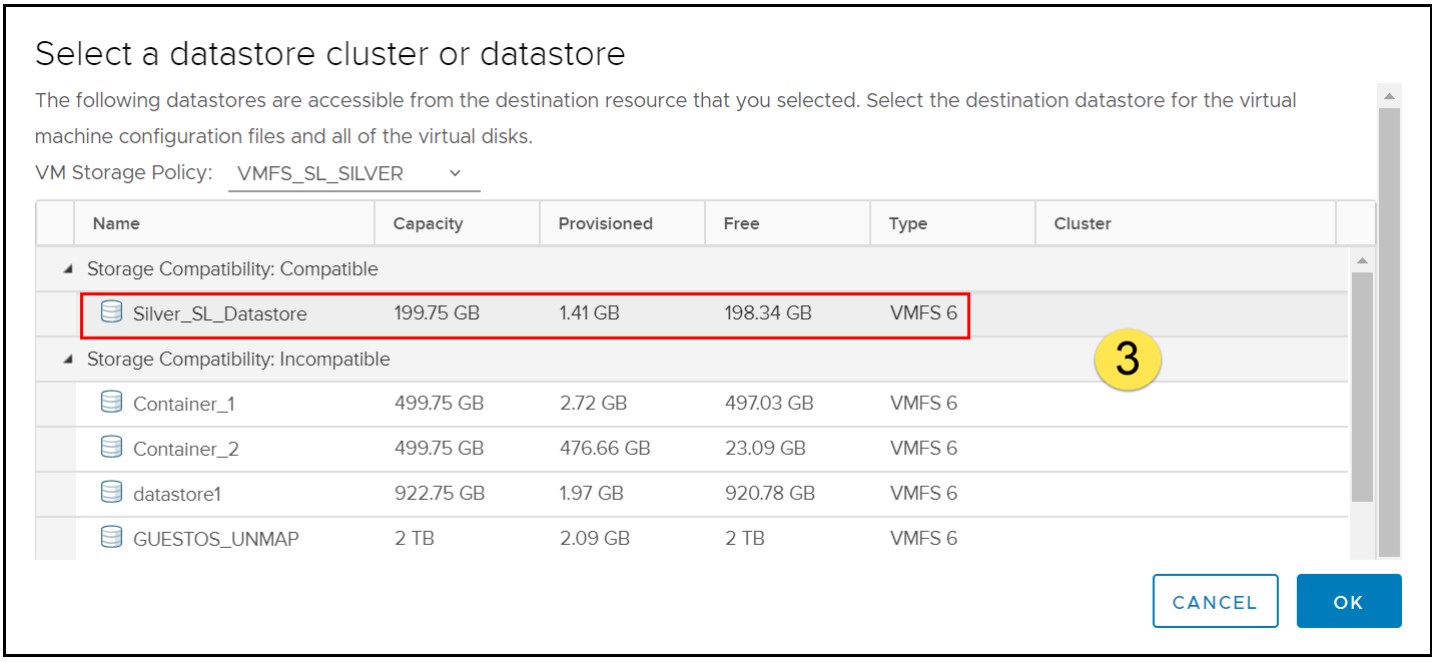

Figure 118. Assigning a new VM storage policy to a virtual machine – step 3

Checking storage policy compliance

Once a virtual machine is associated with one or more storage policies, the VM Storage Policies box will be populated in the summary tab of the virtual machine. This is shown in Figure 119.

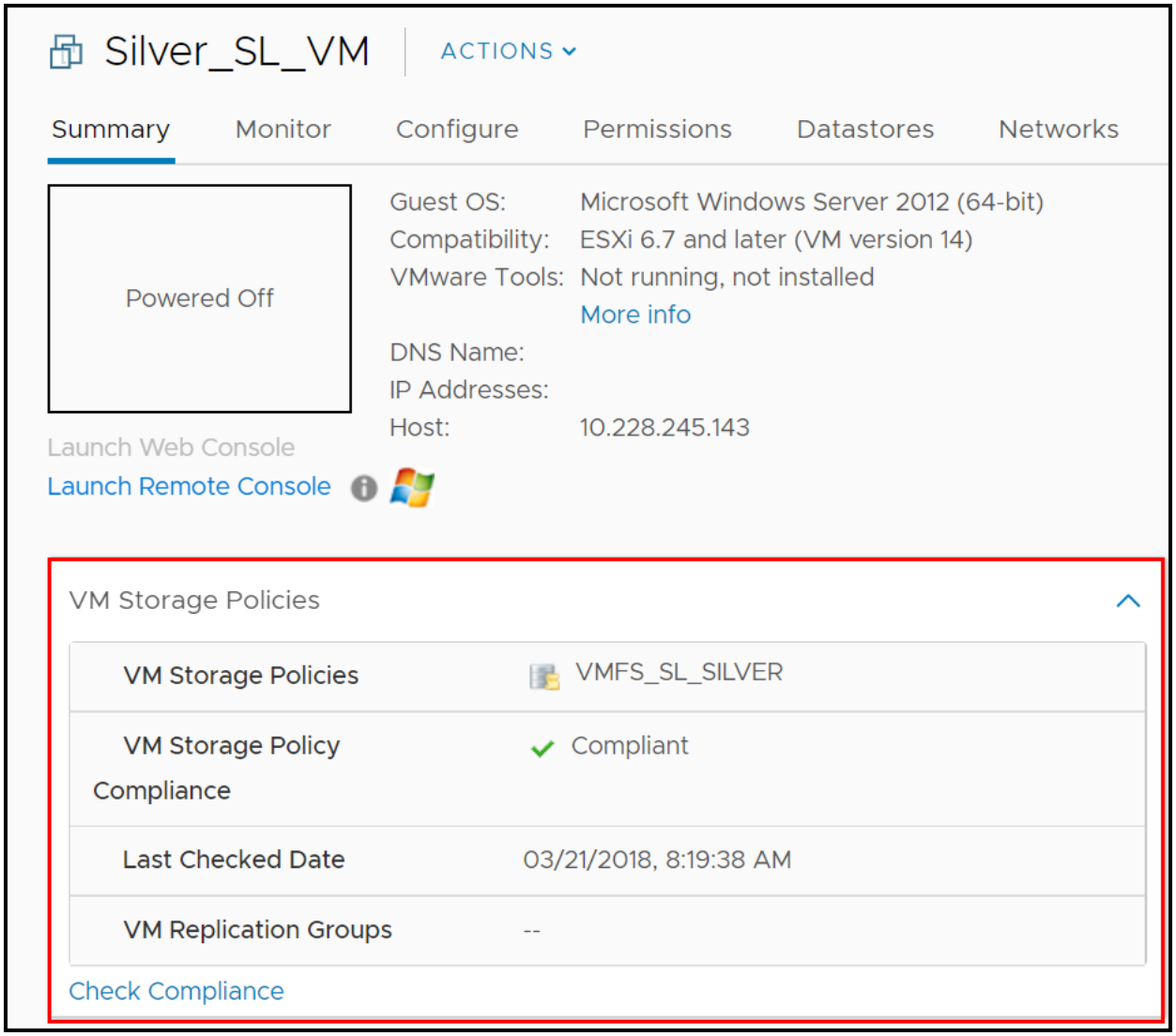

Figure 119. VM Storage Policies box

If all of the virtual disks and configuration files of the virtual machine are compliant, the VM Storage Policy Compliance row will be marked with a green check marked. Otherwise, if one or more virtual disks are non-compliant it will be a red box with a white 'x' (shown in Figure 120). Compliance checking is not real-time so reported compliance information may not always be up to date. It is important to click the Check Compliance link to ensure the correct information is displayed.

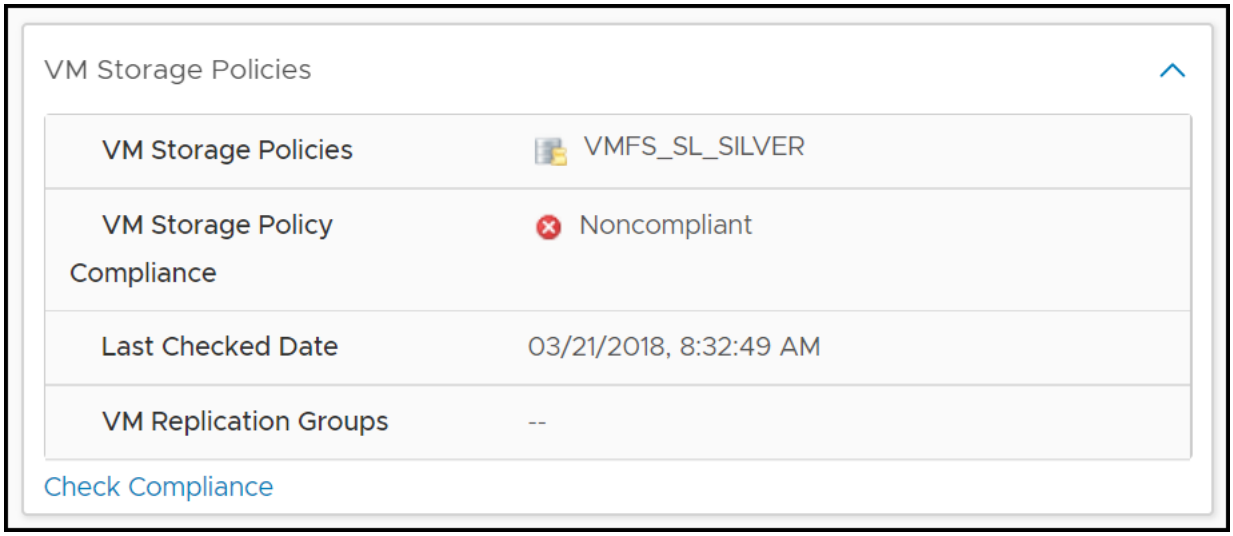

Figure 120. Non-compliant virtual machine

It is also possible to check the compliance of all virtual machines associated with a storage policy by navigating back to the VM Storage Policies view shown in Figure 121.

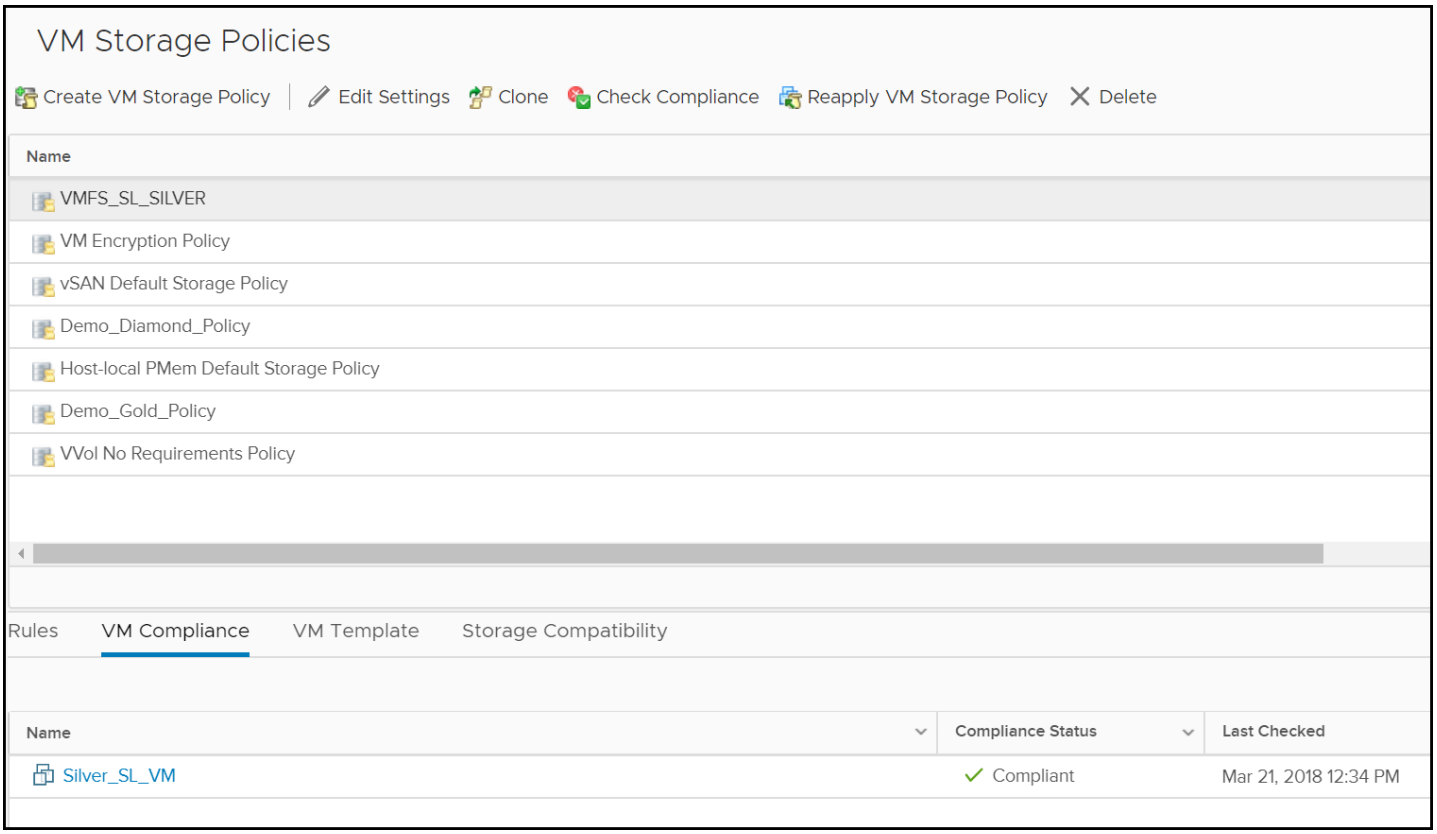

Figure 121. Checking compliance for an entire storage policy

vVol Identification and Monitoring in Unisphere

Identifying vVol WWN in Unisphere

Although vCenter provides no means to map a vVol to the underlying array device, Unisphere for PowerMax does offer this capability. In order to take advantage of the feature, the vCenter involved with vVols needs to be added to Unisphere. This can be done through the VMWARE -› vCenters and ESXi menu in Unisphere seen in Figure 122.

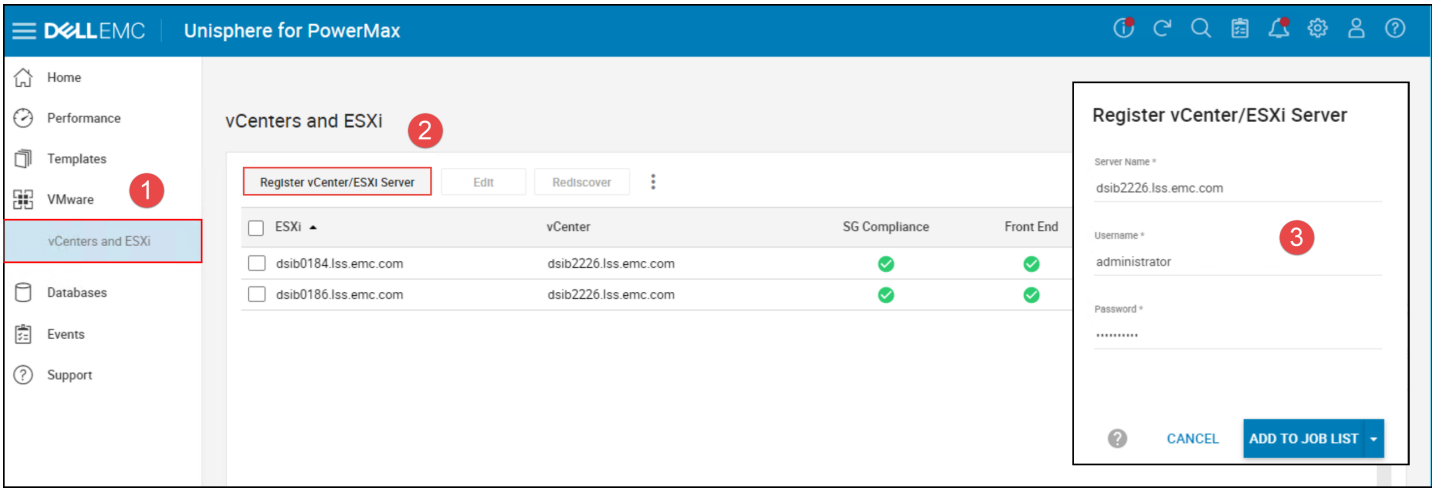

Figure 122. Adding vCenter to Unisphere

Once the vCenter is added, start in Figure 123 by selecting an ESXi host in that vCenter and double-clicking.

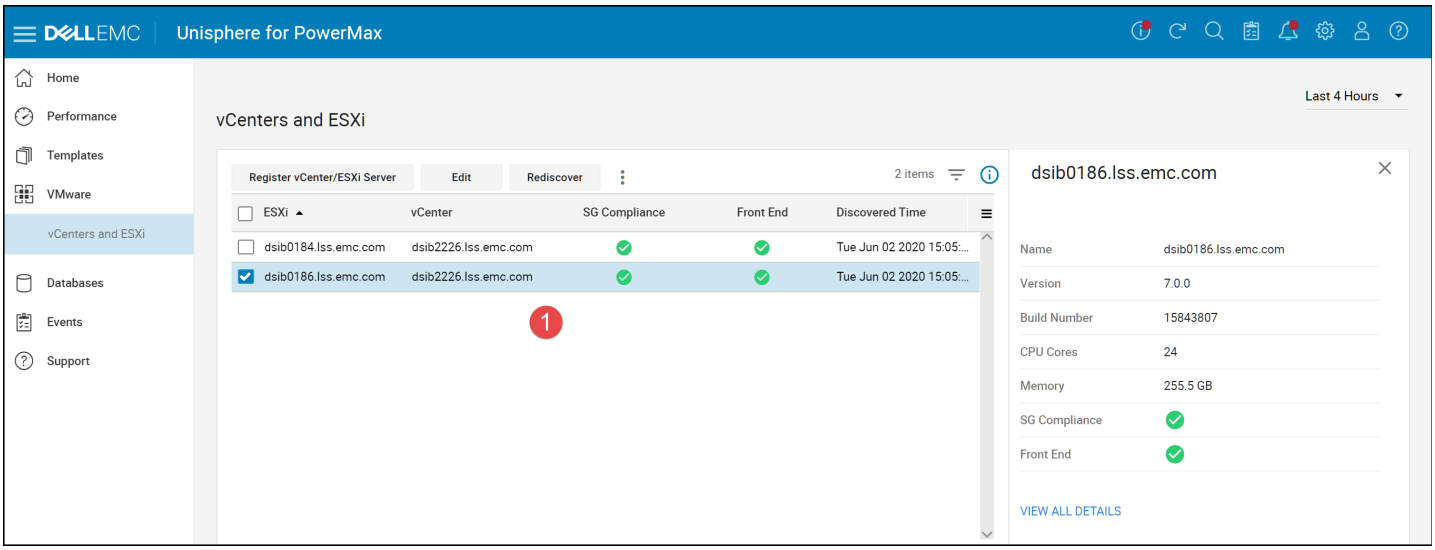

Figure 123. VM vmdk mapping to vVol device - step 1

In steps 2-4 in Figure 124, begin by selecting the Virtual Machines tab, double-click a vVol VM, and in the side panel on the right that will appear, select the hyperlink next to the Virtual Disks row.

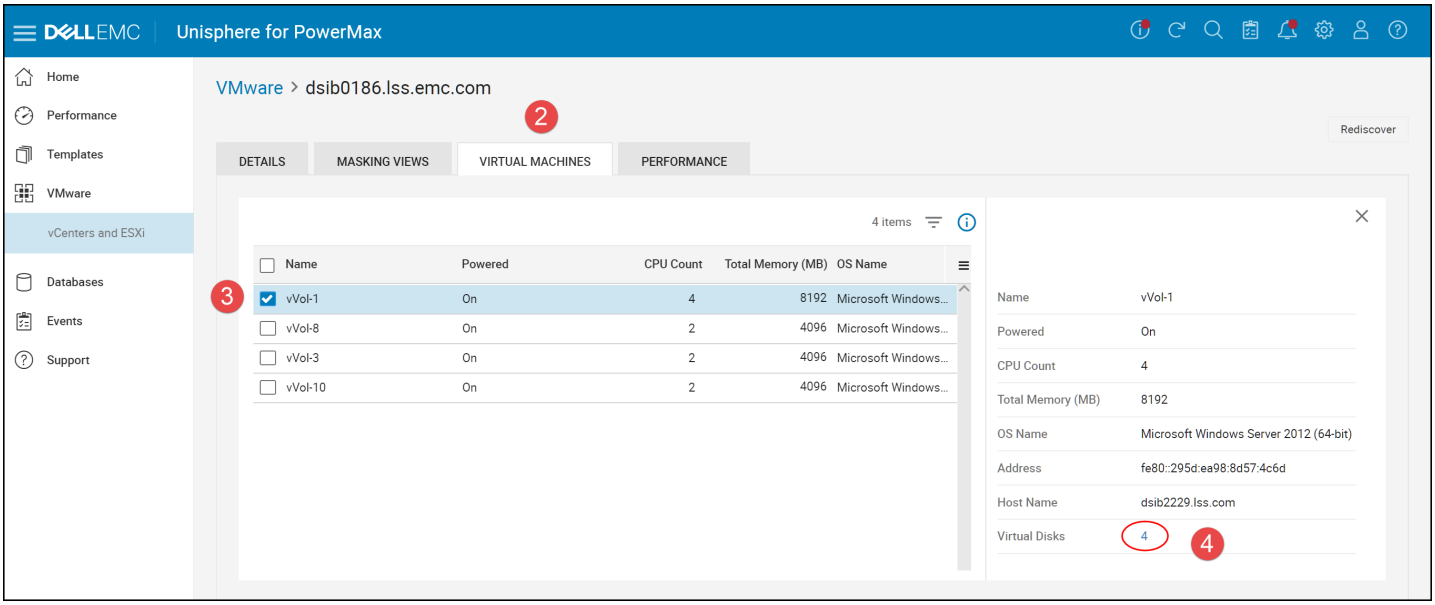

Figure 124. VM vmdk mapping to vVol device - steps 2-4

In step 5 in Figure 125, double-click on one of the Hard disks (vmdk) and in the right-hand panel that appears, all the information about the vVol is displayed. The blue box highlighted in the figure contains the vVol WWN .

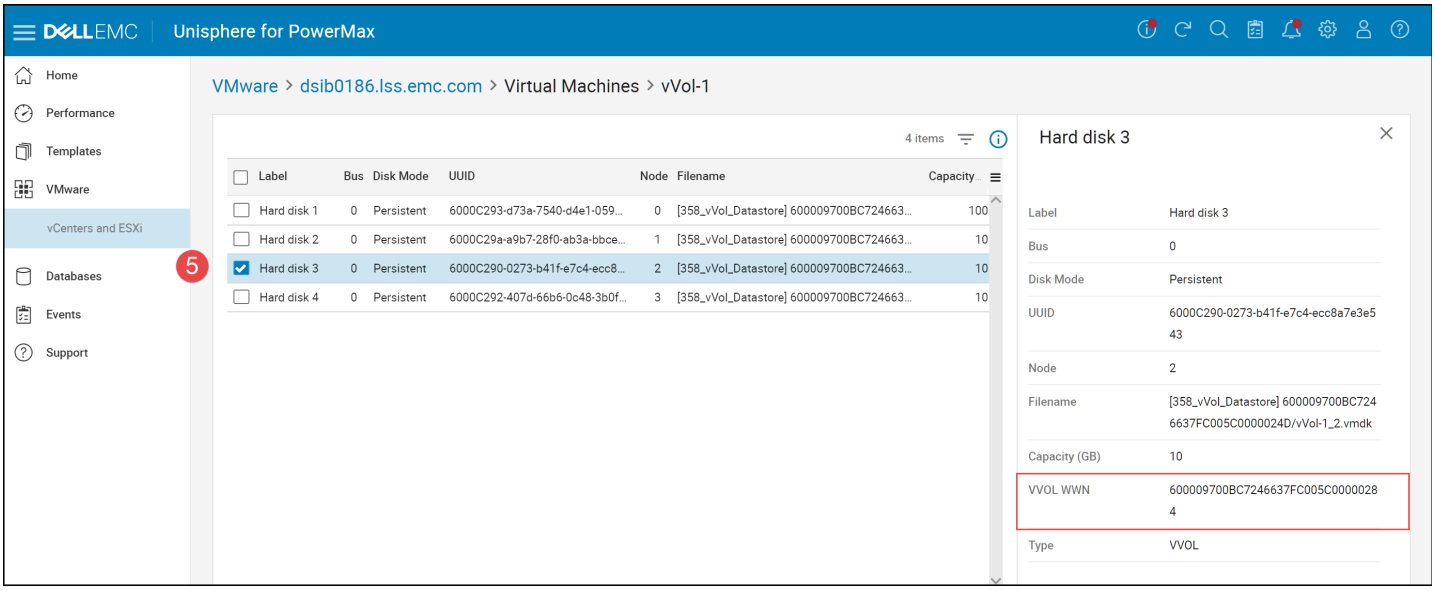

Figure 125. VM vmdk mapping to vVol device - step 5

To determine the device ID, it is necessary to use Solutions Enabler as the ID is not included in the Unisphere output. The command to list the device IDs with the WWN is:

```
symdev list -vvol -wwn -sid 358 |grep <WWN>
```

Using this command, one can see in Figure 126 that the device ID for the vVol in Figure 125 is 001FF.

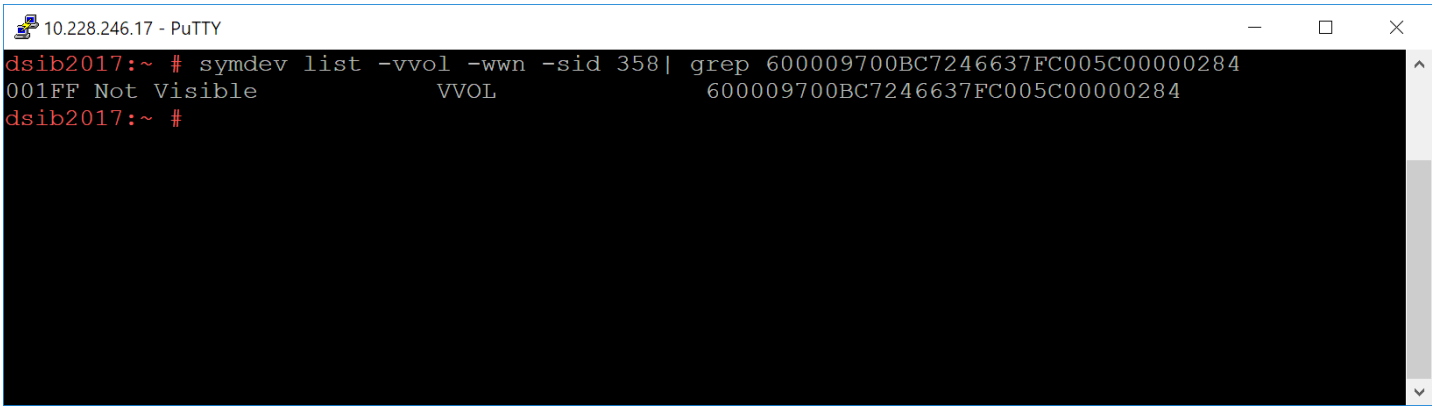

Figure 126. vVol device ID and WWN

In the next section, the device ID can be used to monitor the performance of the vVol.

vVol Performance Monitoring in Unisphere

As vVols on the PowerMax are not visible at the storage group level, performance monitoring needs to be conducted directly at the individual vVol. All metrics that are gathered for regular thin devices are also gathered for vVols. To view performance in Unisphere for PowerMax for vVols, first navigate to the PERFORMANCE -› Charts menu in steps 1 and 2 in Figure 127.

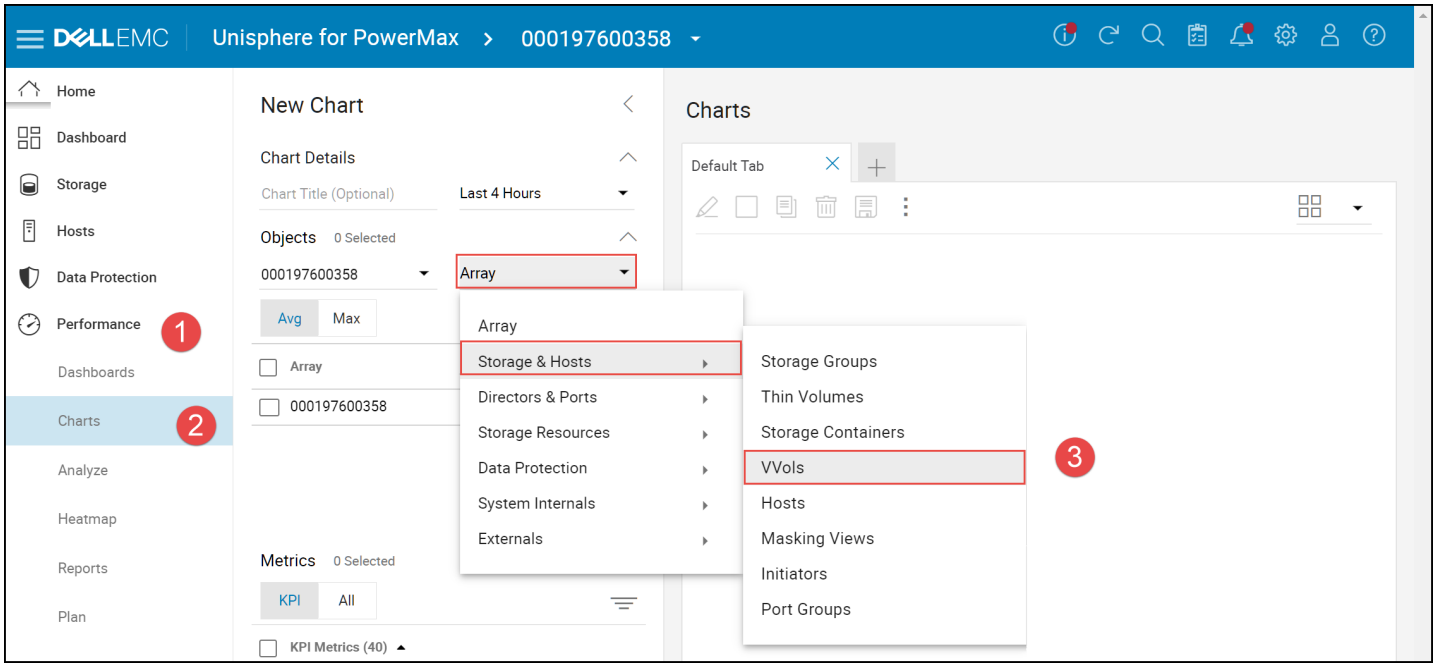

Figure 127. vVol performance monitoring

Next, using the drop-down menu in step 3, traverse to Storage & Hosts -› VVols. In this screen the user can select all, or some vVols, and then a set of metrics and generate one or more dashboards as in Figure 128.

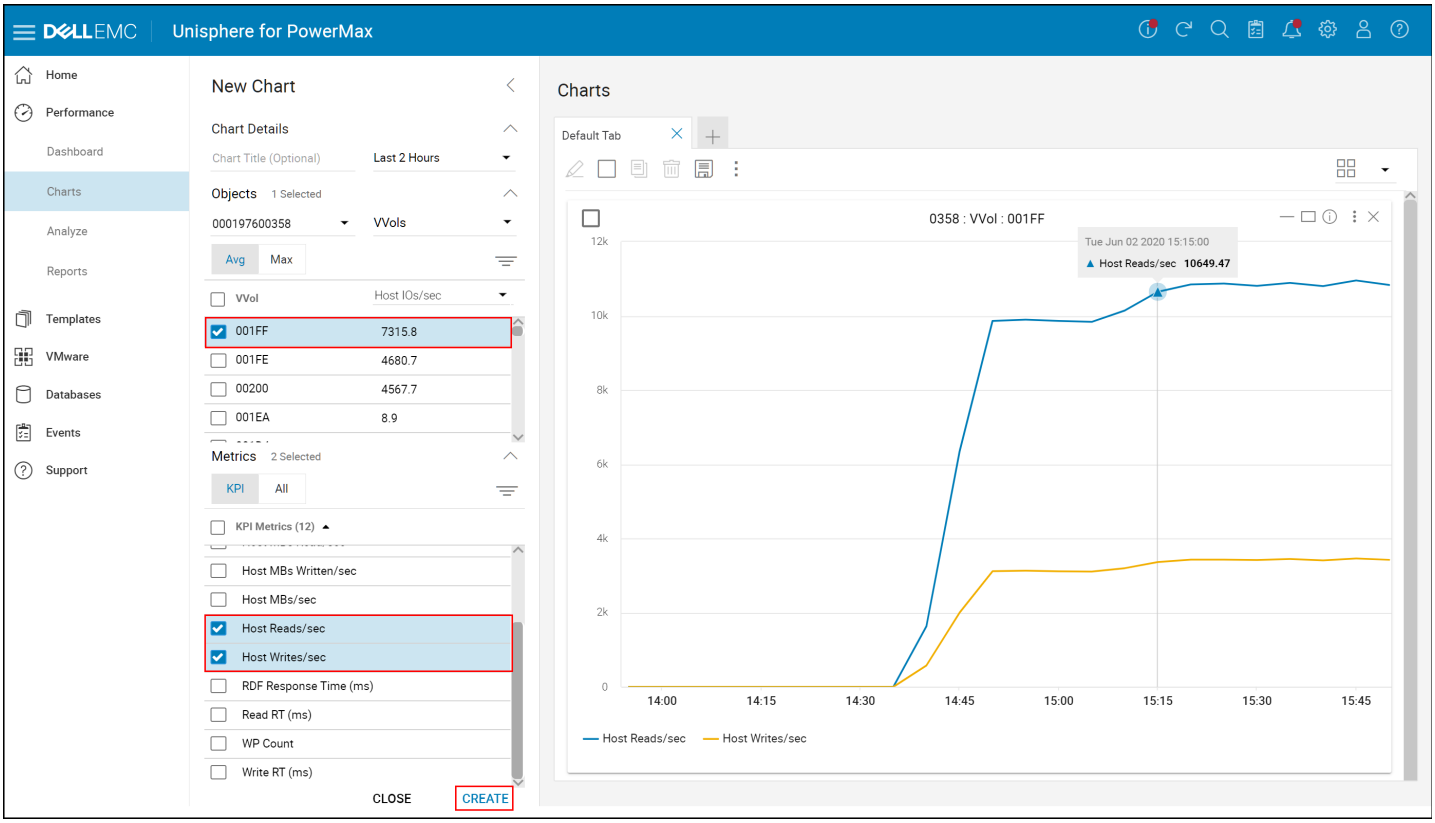

Figure 128. Creating vVol dashboards in Unisphere for PowerMax

In addition to vVol metrics, Unisphere for PowerMax also offers metrics at the Storage Container level. See the Unisphere for PowerMax online help for more detail on creating dashboards.

Scalability

ESXi

With vVols, there is no change to the ESXi device host limit (6.0-256, 6.5-512, 6.7/7.0 -1024). As each VM is made up of many vVols, initially this might seem problematic and prevent scalability; however only the PE device counts against the 256/512/1024 limit. vVols are not mapped and masked directly to the hosts. This permits a vVol environment to scale far beyond a traditional VMFS one, particularly on a VMAX as it supports 64,000 devices.

In addition, a single PE can have a maximum of 16,383 vVols bound to it.

The maximum size of an individual vVol on the VMAX and PowerMax is 16 TB.

Storage Resources

Dell EMC limits the number of vVols that a single storage resource can contain to 4096, or with an Epack[11], a maximum of 8192. Attempts to create any vVols in that resource beyond that value will fail with a general error similar to the following: "*Error: "Cannot complete file creation operation. Operation failed.'"*" It is important to remember that even if the current count of vVols in the storage resource is below the maximum, adding disks or VM snapshots that include memory to VMs in the storage resource, will generate additional vVols. While it is possible to place additional disks to those VMs in a different storage resource, the snapshot vVol will always be placed in the same storage resource. Therefore, in large environments it is important to keep track of the number of vVols in each storage resource. Unfortunately, there is no GUI interface to obtain this information, however the vVol count for a storage resource can be pulled with the CLI.

Utilizing the previously covered commands in the section Using Solutions Enabler with Virtual Volumes, the number of vVols can be pulled from both the storage container and the storage resources within that container. The key to pulling the detail of the number of vVols is to use the output command with the XML format. Figure 129 shows the detail of storage container VVol_Multi_SL_Container using the following command:

```
symcfg show -sc -sc_name VVol_Multi_SL_Container -sid 357 -detail
-out xml
```

The first red box in the figure highlights the name of the container while the second red box details the number of vVols, 15, and snapshots, 4. The xml output does not distinguish between snapshots that include the swap and those that do not. Therefore, the total vVols of 15 may also include the snapshot vVols. As the max vVol limit does not apply to the storage container, each storage resource needs to be examined. The blue and green boxes in the figure contain the vVol counts for each resource, 5 and 14 respectively. The blue and green boxes in the figure contain the total vVols and the total number of snapshots in each resource.

[11] The Epack is already included in the latest PowerMaxOS release PowerMaxOS 5978 Q3 2020.

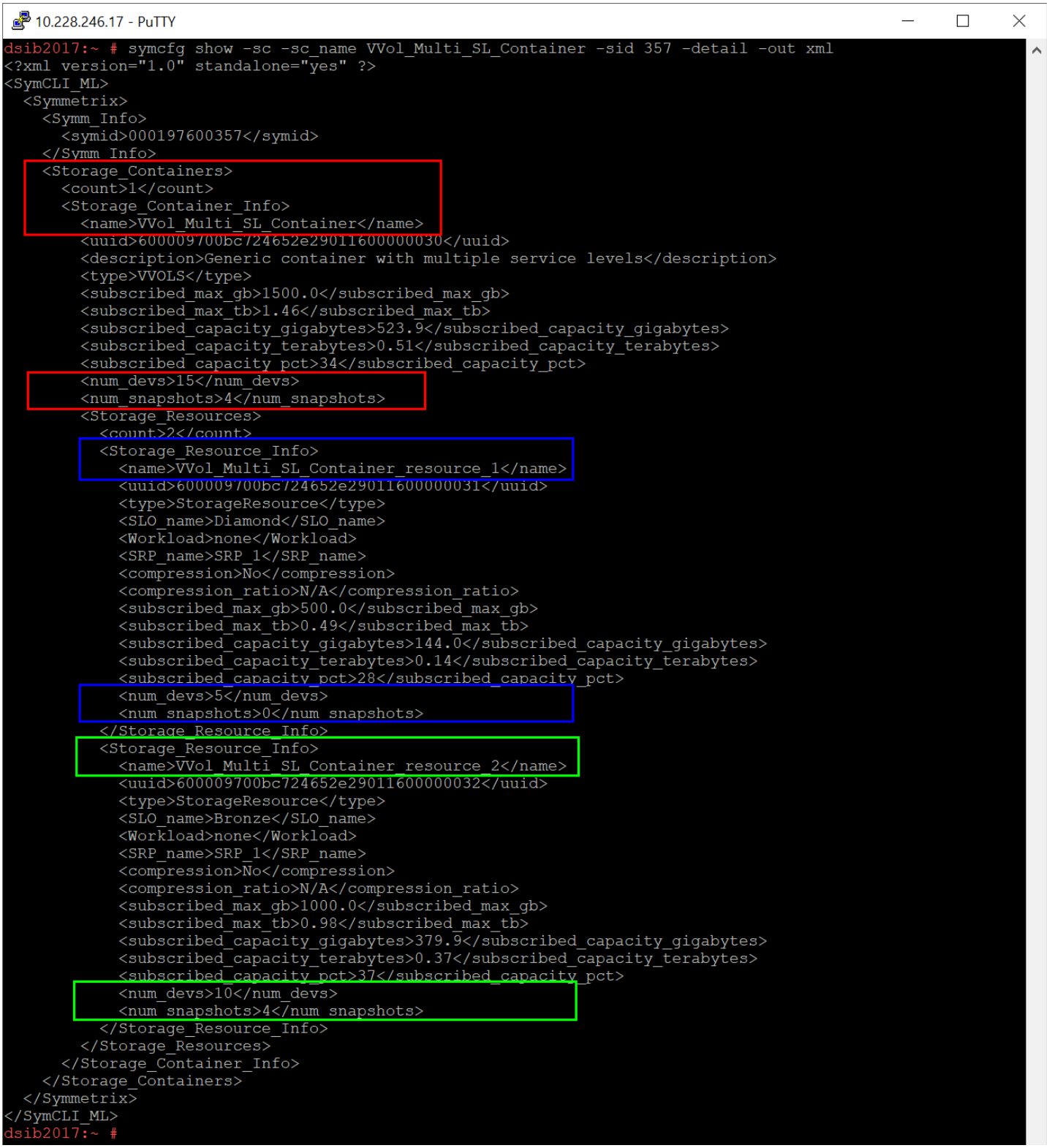

Figure 129. Obtaining vVol count in vVol storage objects

Using this methodology, the vVol count per storage resource can be managed.

Since each storage container can only have a single storage resource of a particular service level, if a resource approaches the 8192 count it may be necessary to create a new storage container with a new storage resource of that service level. VMs or individual vmdks can then be moved to the new container.

An Epack is required to change the vVol storage resource maximum from 4096 to 8192 prior to PowerMaxOS 5978 Q3 2020 (5978.669.669). It can be requested via support.

VM snapshot sizing

One of the benefits of vVols is the ability to use the array data services directly. Most of these are applied through Storage Policy Based Management (SPBM), but the most direct integration is snapshots. When taking a snapshot of a vVol-based VM, VMware passes off control to the array to take the snapshot using, in this case, TimeFinder technology. Array snapshots are incredibly efficient, because they are targetless. Unlike traditional VMFS, there is no requirement to keep delta files on the file system. All deltas are kept on the array. In addition to the targetless snapshot, if the user wishes to capture the memory state of the VM, VASA generates a static vVol with that information, equaling the size of the VM memory.

Since tracks changes for snapshots are stored on the array, there is some storage usage. The question then is how is this accounted for? Is the usage deducted from the storage container/resource? Yes and no. The following example will explain the process.

Setup

A single storage container is created, Demo, which contains two storage resources in Figure 130:

- Demo_resource_1 is 1000 GB with an SL of Gold
- Demo_resource_2 is 1000 GB with an SL of Silver

Therefore, there are two storage policies, one for Gold and one for Silver.

There is a single 100 GB vVol VM, i.e. a single 100 GB vmdk, created with a Gold storage policy. In the vVol datastore (the Demo storage container) there are 3 vVols – 1 – 4 GB config vVol, 1 – 8 GB swap vVol, and 1 – 100 GB data vVol, or 112 GB of allocated vVols. These are the storage resources in Figure 130 listing the amount subscribed.

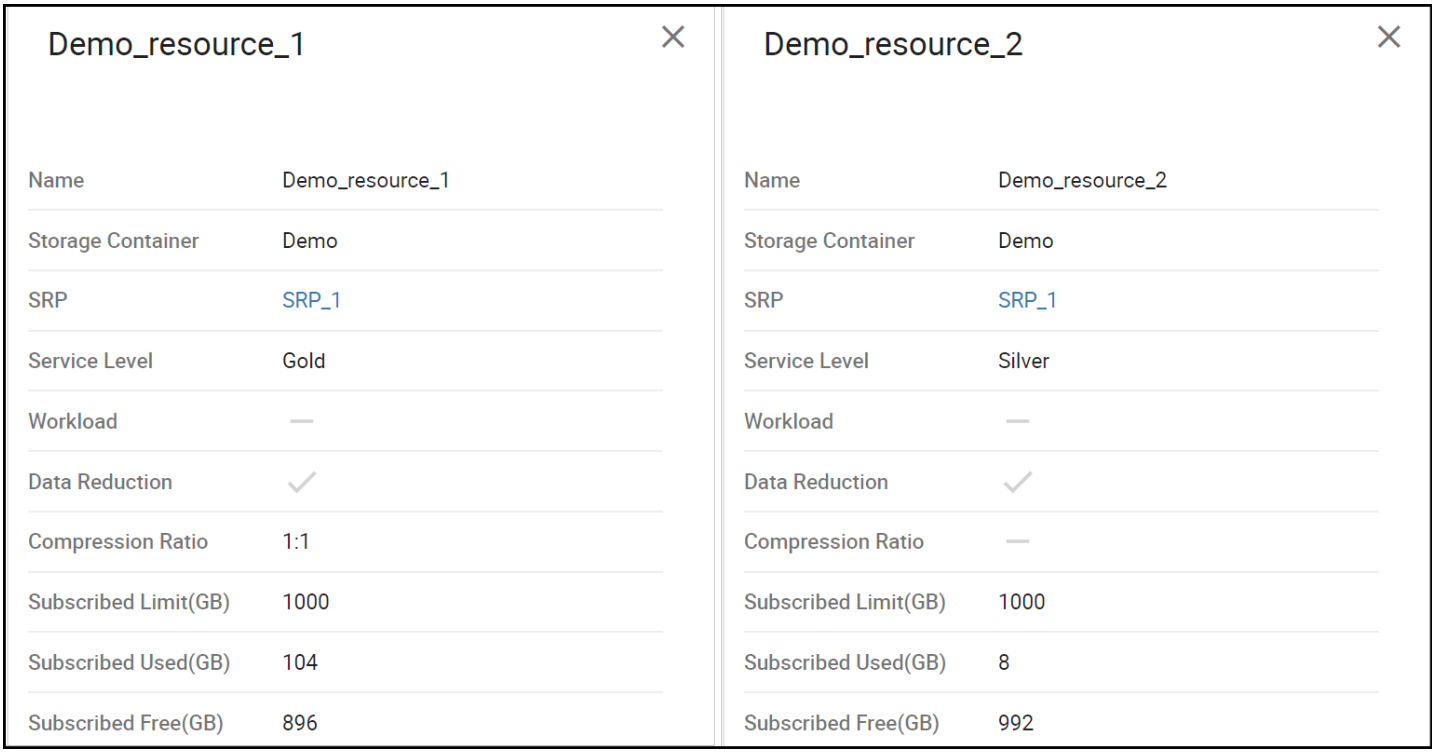

Figure 130. Storage Resources

After reviewing one may wonder why there is only 104 GB subscribed in the Gold storage resource when the expected total would be 112 GB. The vVol implementation is designed such that the swap vVol will always be placed in the lowest performing service level in the container, so in this case the 8 GB swap is in the Silver storage resource. As the data associated with that vVol is in memory on ESXi, the idea is to not waste space in the higher performing service levels.

Snapshot preserved space

Targetless snapshot

The first step is to take a targetless (no memory) snapshot of the VM in Figure 131.

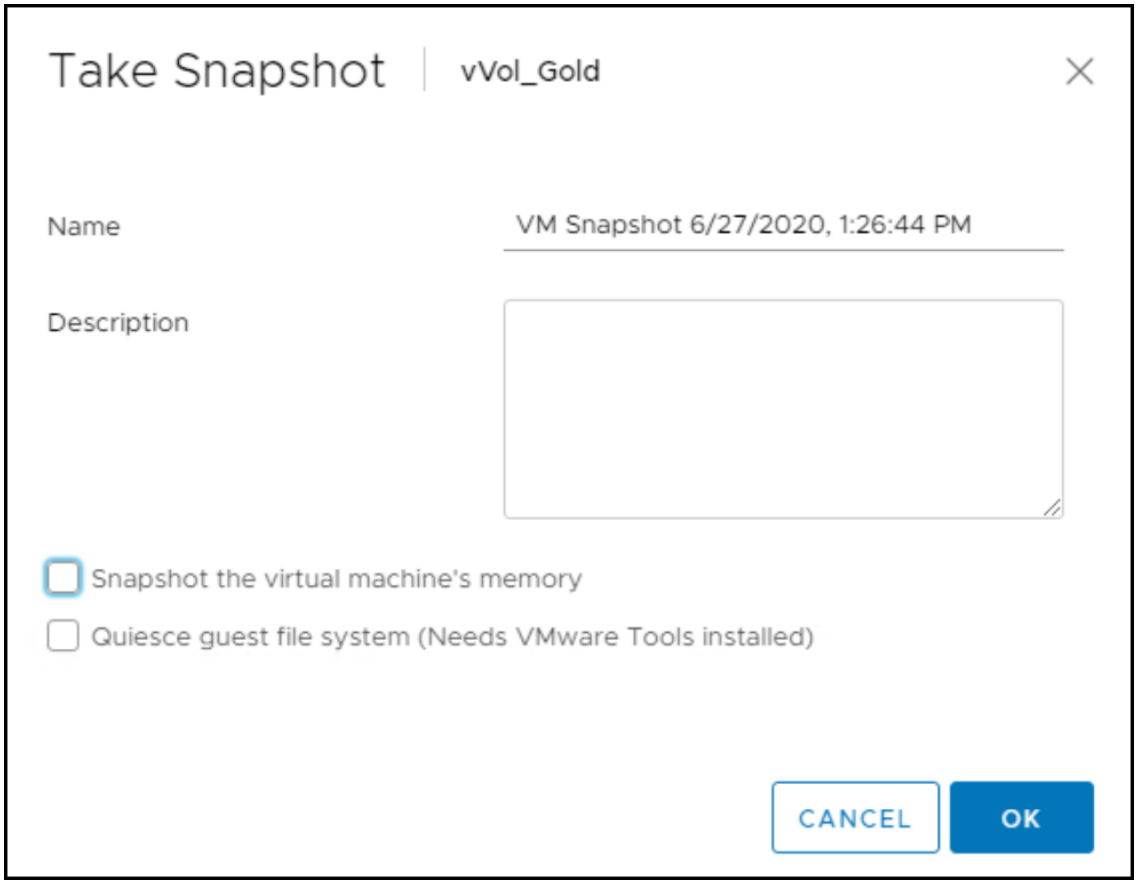

Figure 131. Targetless snapshot

It finishes almost instantaneously since the array is doing it. Once it completes, the summary information for the VM in Figure 132 will show that one snapshot is taken, utilizing 100 GB.

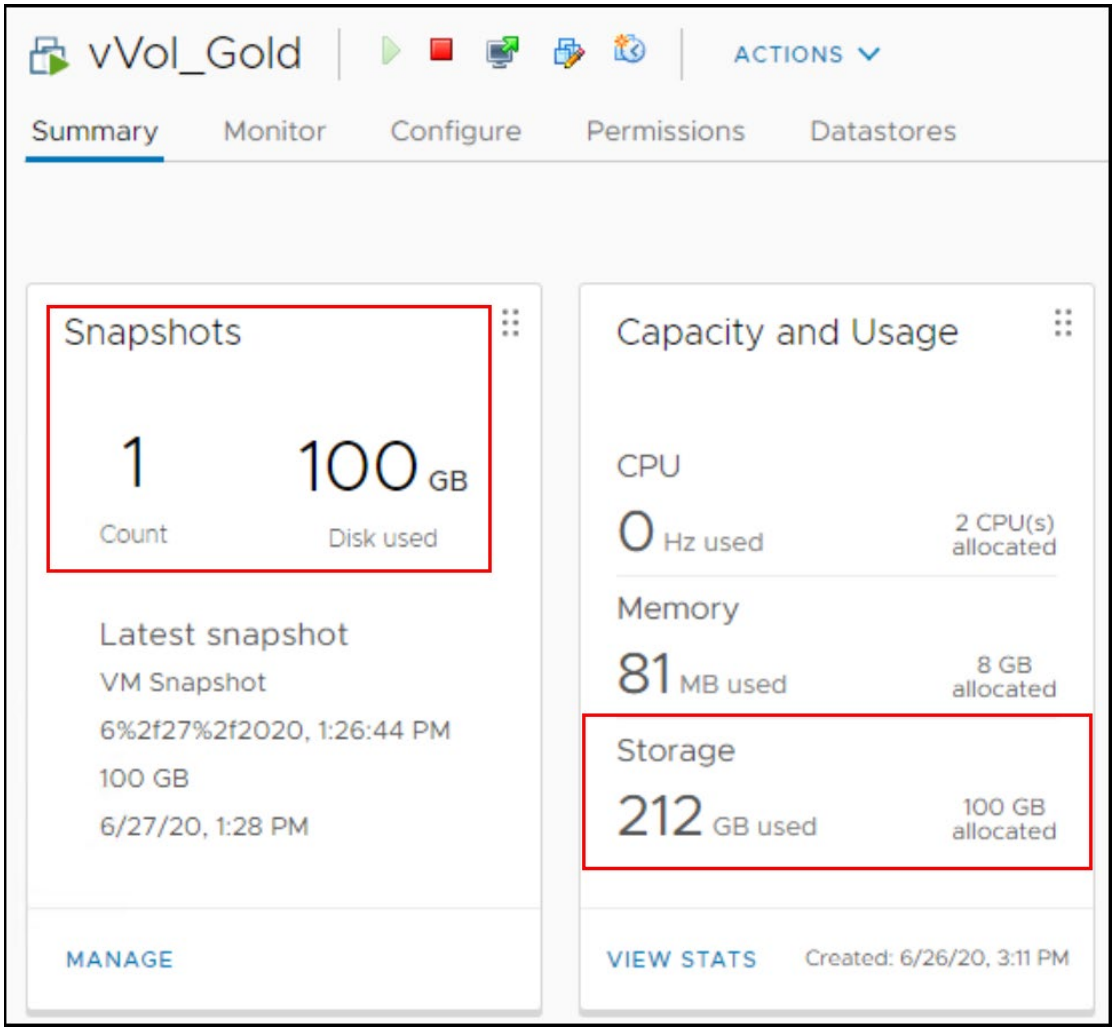

Figure 132. VM summary

VMware records the single snapshot of the data vVol (100 GB) and shows that the VM is using a total of 212 GB (note that VMware shows only 100 GB of that is allocated since the other 100 GB is the snapshot). Reviewing the storage resource on the array in Figure 133, it now indicates there is 204 GB subscribed.

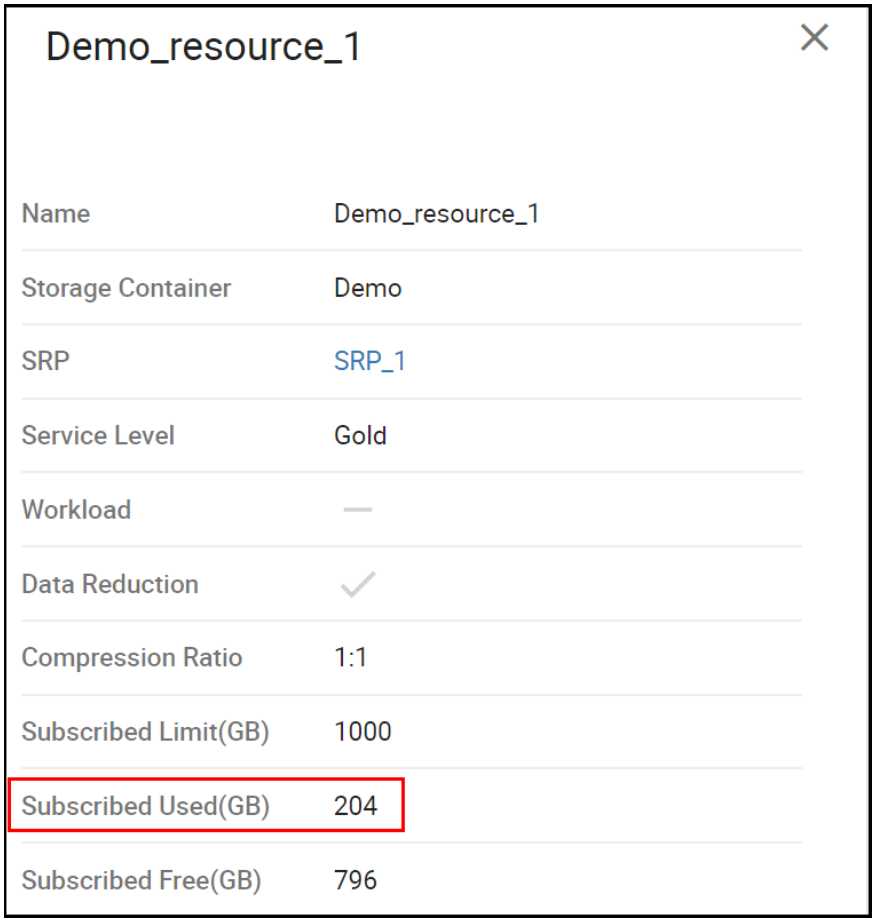

Figure 133. Storage resource allocation without memory

The array has reserved that 100 GB of snapshot space that it is not associated with a vVol. The reservation is a preventive step for future restores. Since SnapVX technology is utilized on the array, the first thing the array will need to do for a restore is create a target device to link the snapshot to. If the space is not reserved up-front, there is always the potential that the storage resource could be out of space by the time the restore is run, and thus the restore would fail.

Memory snapshot

In step 2, take a memory snapshot. Note that the storage usage in Figure 134 is now at 320 GB, not 312 GB, because of the 8 GB of memory.

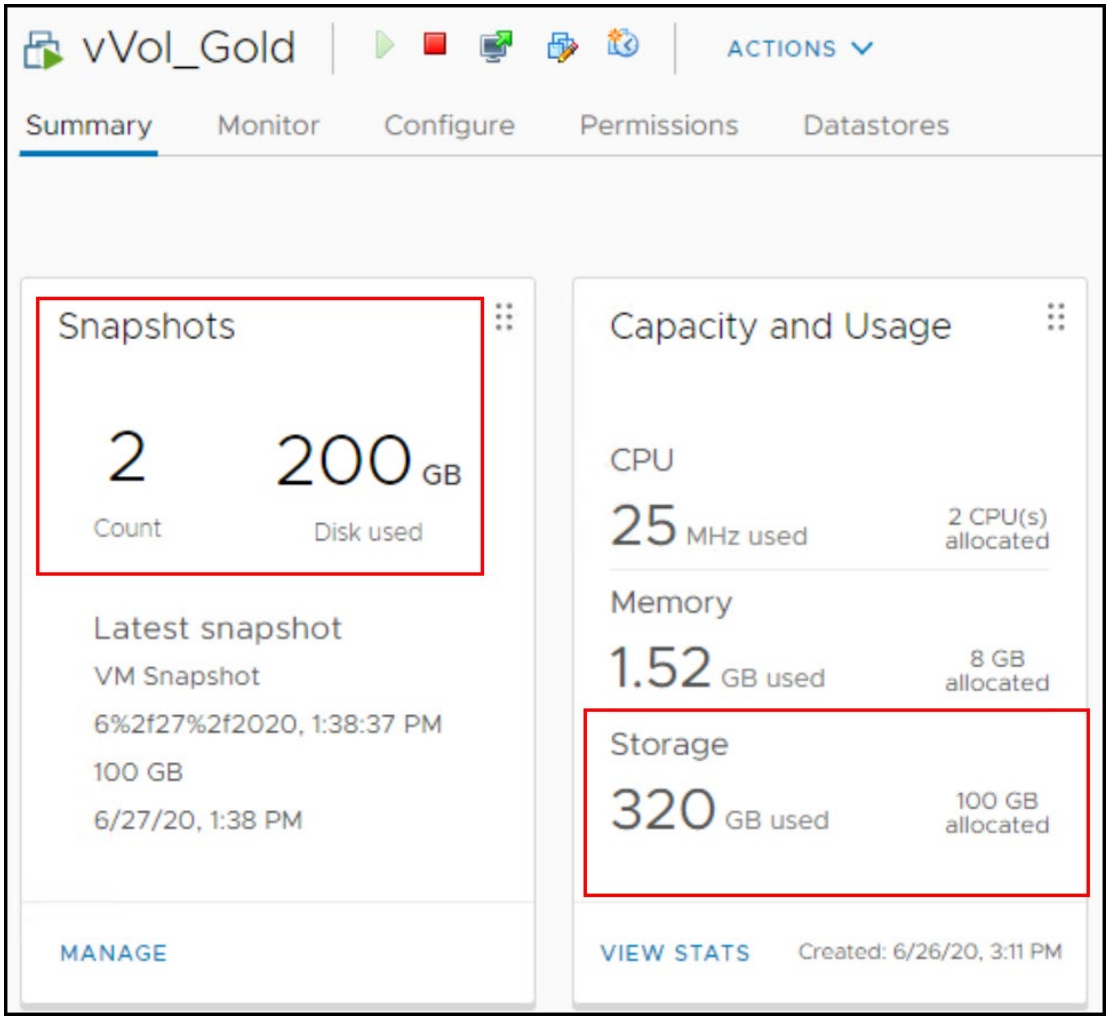

Figure 134. Memory snapshot

However, in Figure 135 see how the memory snapshot vVol is stored in the Silver service level, not Gold, as Silver is the lower tier.

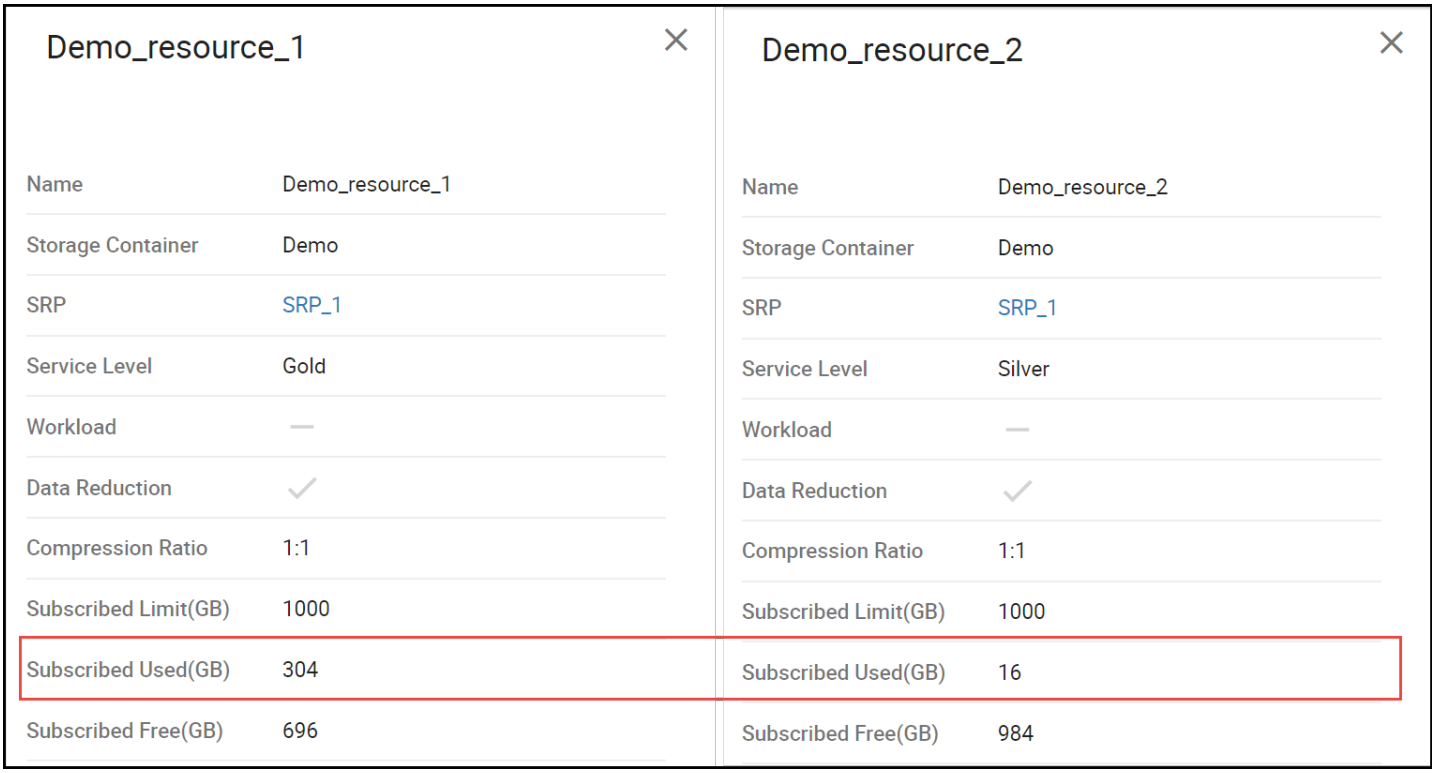

Figure 135. Storage resource allocation with memory

Overallocation

The creation of the swap file is one of those tasks that will not fail due to lack of space. For example, if this VM is powered down, and then the capacity of the Silver resource adjusted down to 10 GB so that the 8 GB swap will no longer fit, VASA will over-allocate the resource to prevent failure as in Figure 136.

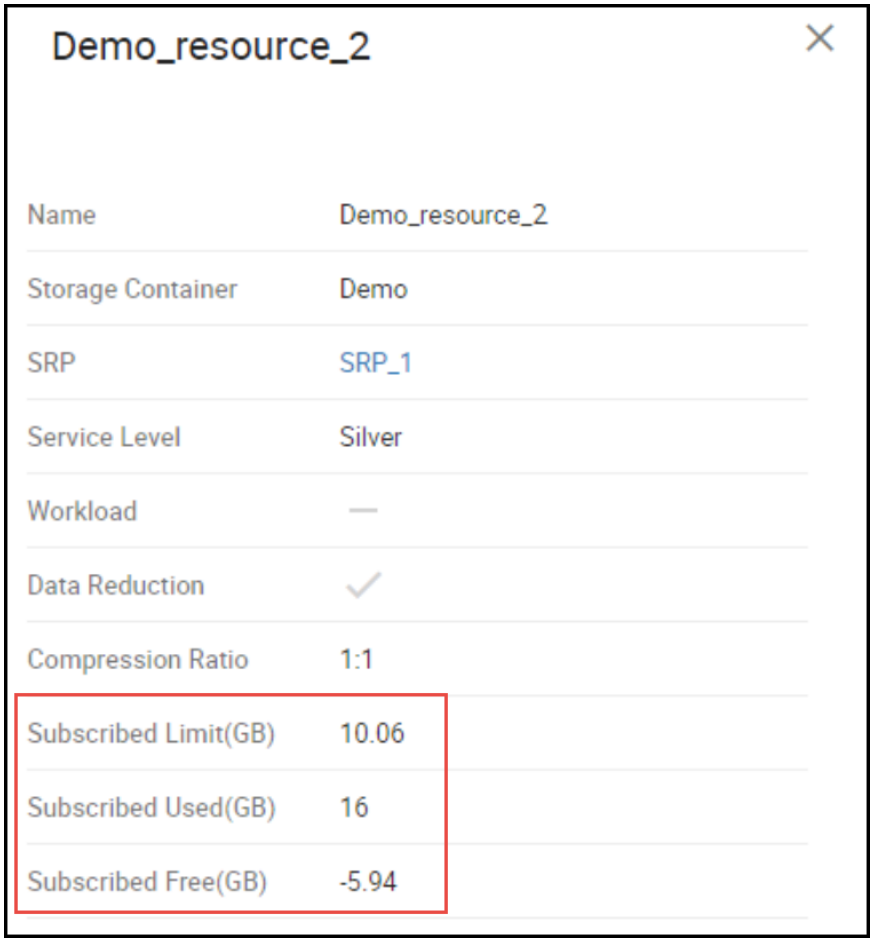

Figure 136. Overallocation of the storage resource

Once a storage resource is oversubscribed, no additional files will be placed there.

Storage Demand Report

Although a Storage Container does not reserve space in the SRP, the active usage of the container is accounted for at the storage group level. Keeping track in the storage container would be resource intensive; however, it is still possible to determine the snapshot usage in the SRP for those storage resources. Within Unisphere for PowerMax there is a Storage Demand Report which shows how much each storage group is using, including snapshot space. Each storage resource in a storage container is represented by a storage group.

First navigate to the main dashboard of the array and select “Capacity” in Figure 137.

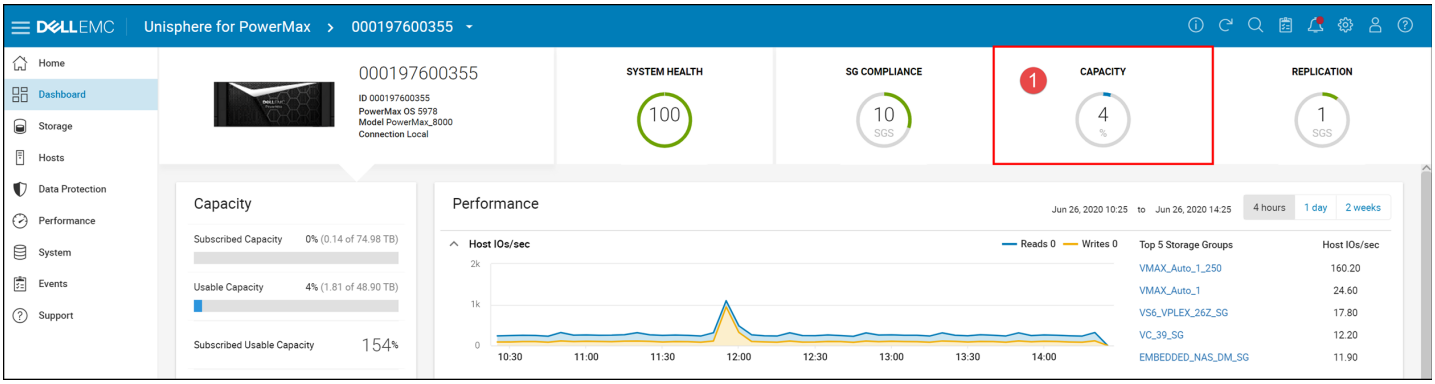

Figure 137. Storage Group Demand Report - step 1

In the second step, use the System drop-down to select the correct SRP. Then in step 3 select STORAGE GROUP DEMAND Action. These are outlined in Figure 138.

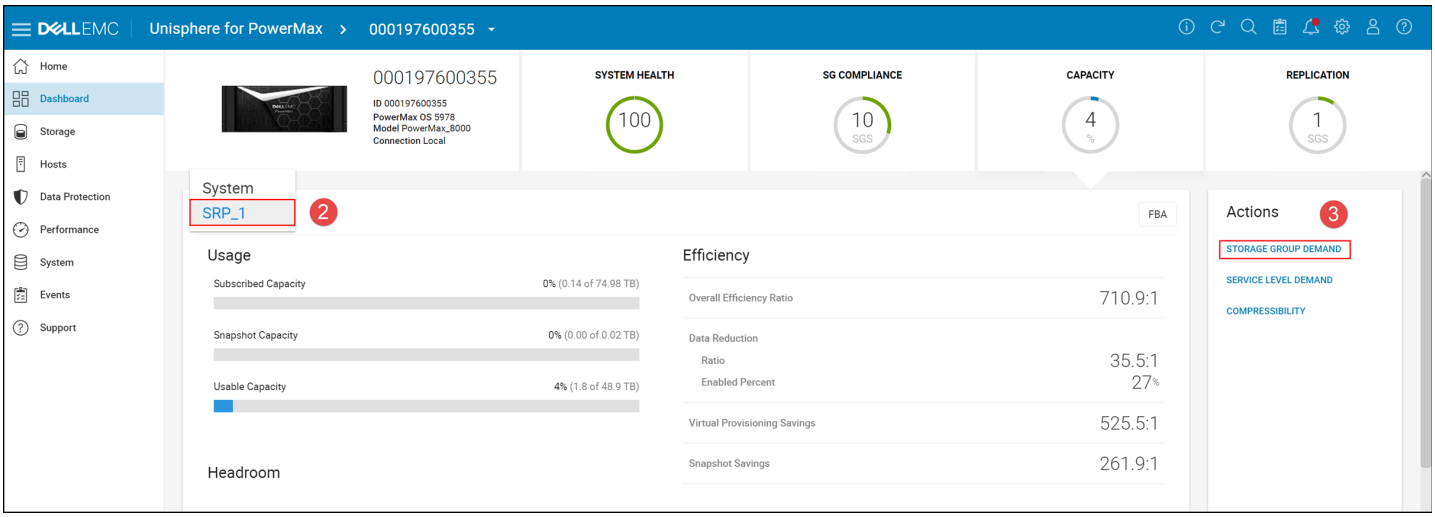

Figure 138. Storage Group Demand Report - step 2 - 3

The Storage Group Demand Report in Figure 139 will list all storage resources in the container as storage groups with the prefix "_VVOLS_". Here one can see the snapshot information for two storage resources. Again, the snapshot data listed here will not be accounted for in the storage resource, just the SRP. Also note that this report does not include the snapshot reserved space, since that is only reserved in the storage resource and is not represented by vVols in a storage group.

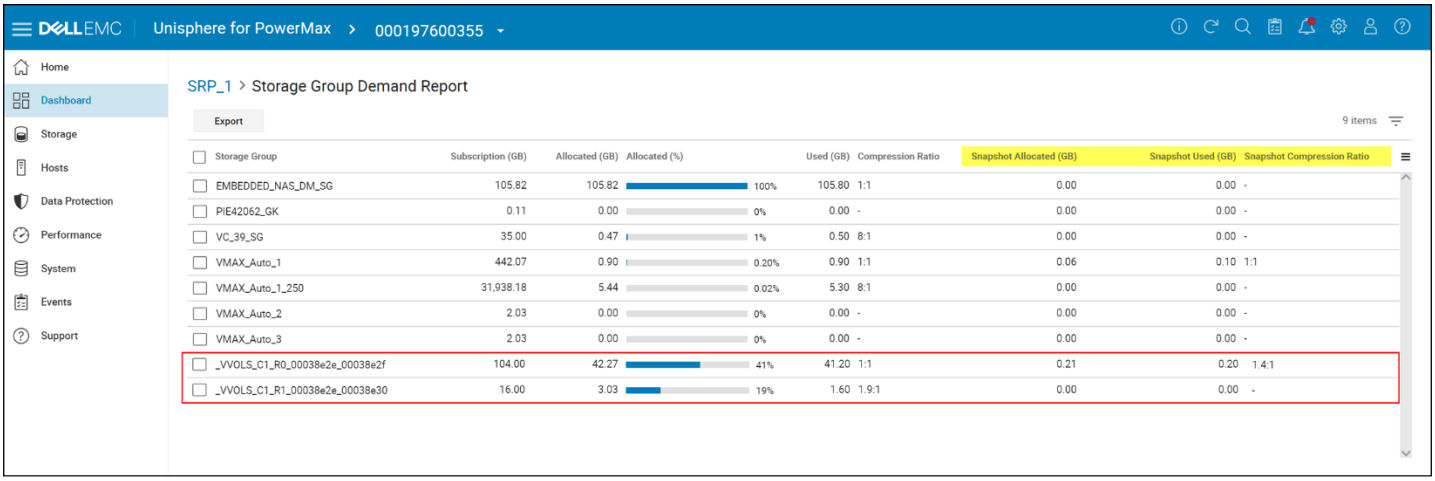

Figure 139. Storage Group Demand Report

Unfortunately, the naming of the storage groups for vVols is not straightforward. In fact, it looks downright cryptic:

- _VVOLS_C1_R0_00038e2e_00038e2f
- _VVOLS_C1_R1_00038e2e_00038e30

Fortunately, it can be decoded using the *symcfg* Solutions Enabler command by outputting the storage container in XML format. The XML format will give extra details about the container and its storage resources. Part of the detail is the UUID of the container itself and each storage resource. This is how the vVol storage group is constructed. The format is "_VVOLS_C" then the number for the order in which the container was created starting with 0. In this case the demo container is the second one, so it gets the number "1". Next is "_R" followed by the resource number, again in the order created. The final two parts are comprised first of the last 8 characters of the container UUID then the last 8 characters of the resource UUID. The values are highlighted in Figure 140, but decoding:

- _VVOLS_C1_R0_00038e2e_00038e2f – Gold storage resource
- _VVOLS_C1_R1_00038e2e_00038e30 – Silver storage resource

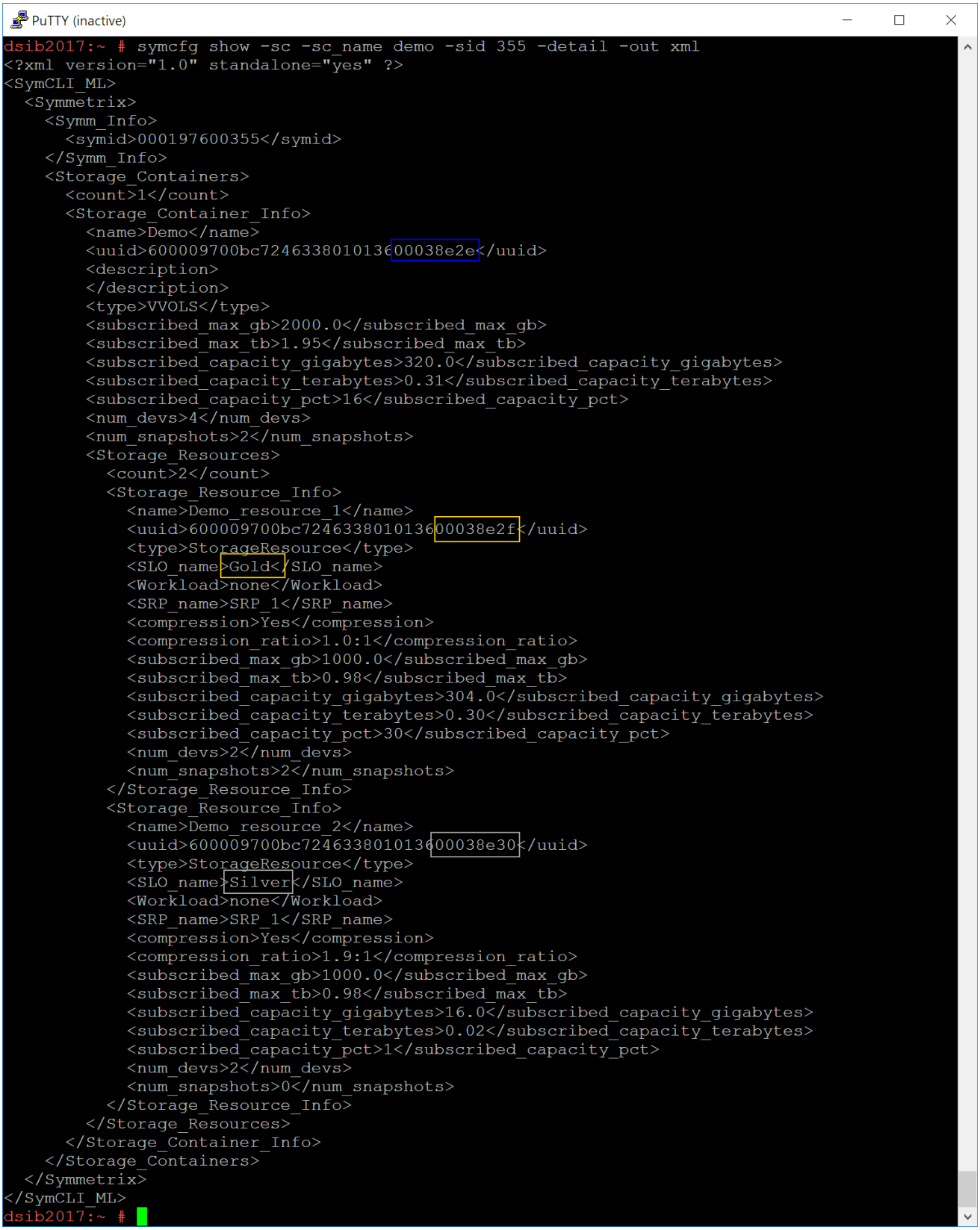

Figure 140. XML output of storage resource

The XML output also has important information about the container. In particular it will show you how many vVols are in each storage resource. For this example, there are 2 vVols in the Gold resource and 2 snapshots, while in the Silver resource there is also 2 vVols, those being the swap (memory) vVols. Additional information about this command can be found in the section Storage Resources.

Solutions Enabler Storage Group Demand Report

If Solutions Enabler is preferred, the following command will produce the same results as the Storage Group Demand Report. The command in Figure 141 is:

```
symcfg -sid xxx list -demand -srp -type sg -detail -gb
```

Note the image below is not from the same array as Figure 140.

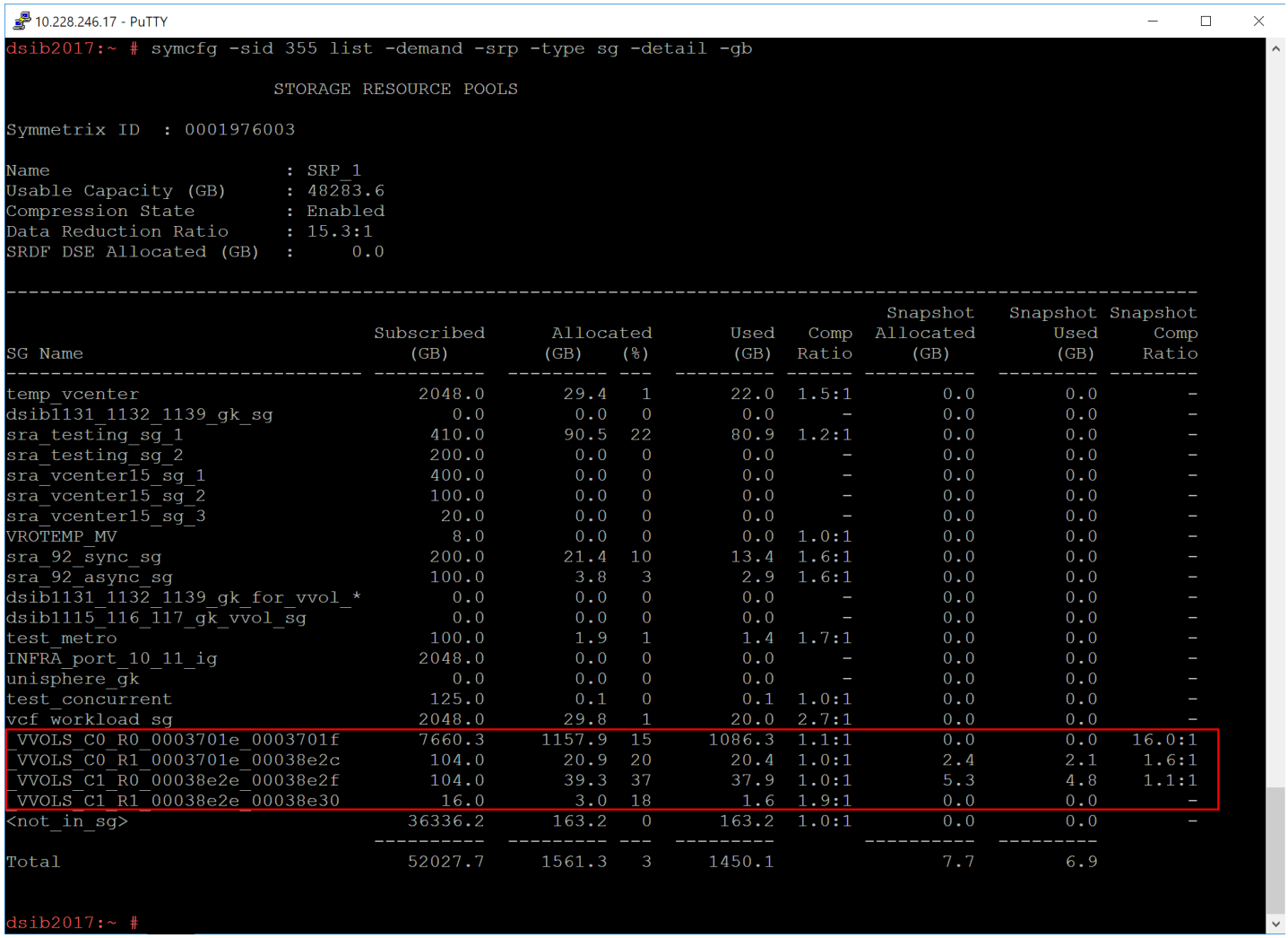

Figure 141. Generating the Storage Group Demand Report in Solutions Enabler

Queueing

As each ESXi host only has a single PE for all IO, customers may be concerned that queuing could become an issue. A PE has a default queue depth limit of 128,

considerably higher than a normal VMFS or RDM device which defaults to 32 (max of 256), but it does represent all vVols on that host. The question then, is how much IO can that one PE handle?

The following example will illustrate how much IO is possible using the default settings. The VMAX and PowerMax arrays are all flash arrays, meaning their latency is very low, usually sub 1 millisecond. For this example, the latency will be rounded to 1 millisecond. If one assumes 1 millisecond this means 1000 IOs can be completed in a single second (1000 milliseconds in 1 second). Therefore with 1 outstanding IO in the queue 1000 IOs can be serviced. Now while the PE queue defaults to 128, if the HBA device queue depth limit is lower than that, and with fibre channel it usually is, it will be the actual queue size the vVols use. Using the previous calculation and assuming an average default HBA device queue depth limit of 64, this means 64,000 IOPS are possible to that one PE. If the HBA queue equaled the default PE queue, it would be twice that, 128,000 IOPS. While that is considerable for a single host, if the customer environment requires more IOPS the following section explains how to adjust the PE queue.

Adjusting the PE queue

The parameter which controls the PE queue is Scsi.ScsivVolPESNRO and is shown in Figure 142.

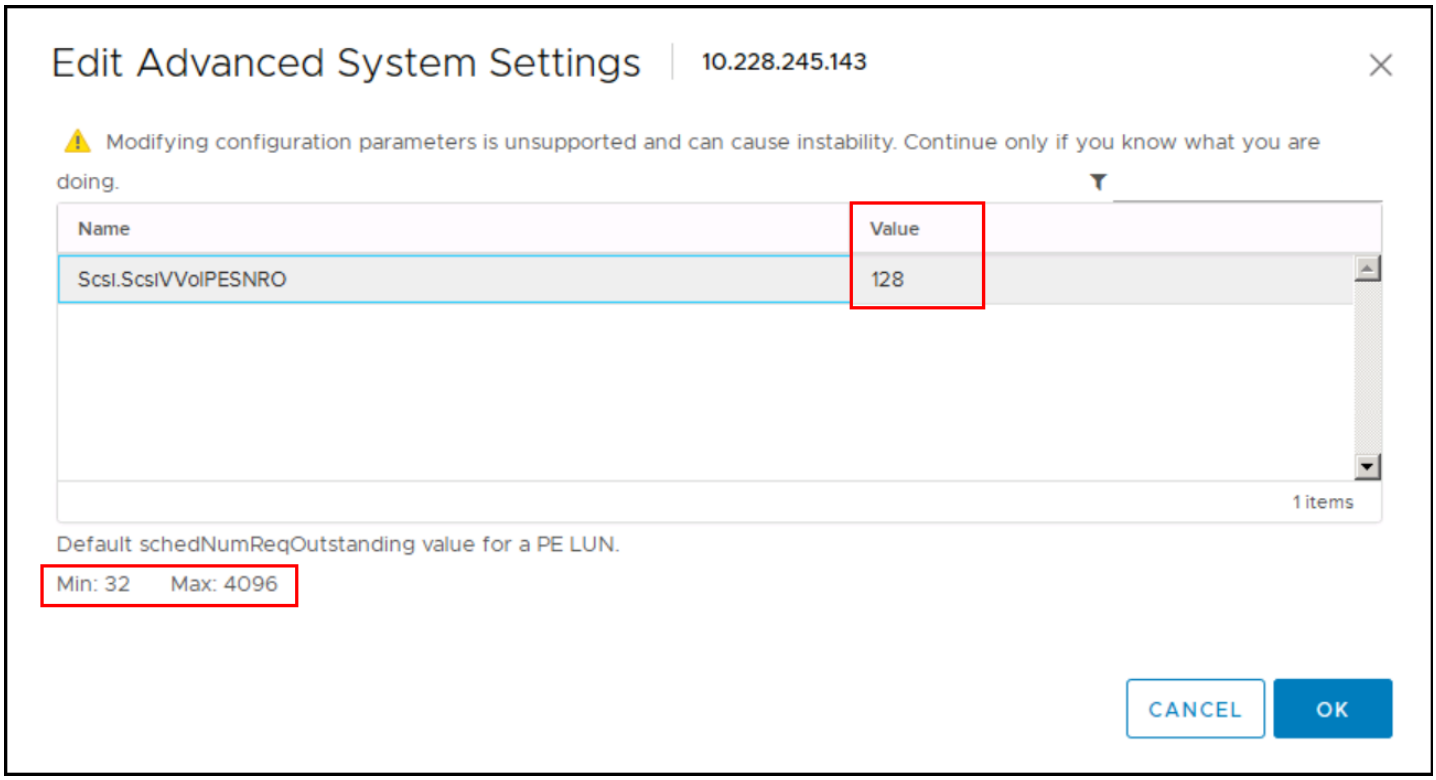

Figure 142. Adjusting the PE queue

While the PE queue depth limit can be changed in the Advanced Settings, the new value will not take effect until the next reboot. In addition, it is important to remember that having a PE queue depth limit larger than the HBA device queue depth limit will

not make any difference, though there is no concern leaving the PE at the default if it is larger.

If a reboot must be avoided, the PE queue depth limit can be altered through the following command line:

```
esxcli storage core device set -O <number> -d <naa>
```

There are a couple caveats about using the command line for this change. First, in vSphere 6.0, the number cannot be larger than 256. Anything higher will generate an error indicating 256 is the maximum. Second, in vSphere 6.5 while the value can be changed to 4096 through CLI, the number cannot be larger than the current device queue limit or an error will be generated to that effect as in Figure 143.

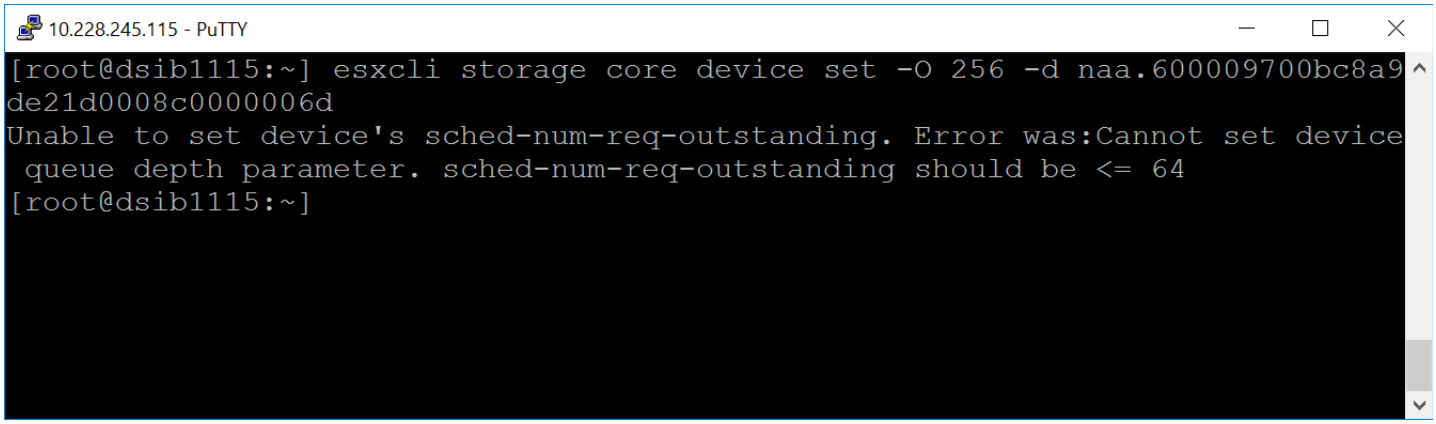

Figure 143. Attempting to set the PE queue beyond HBA device queue depth

A change to the HBA device queue depth limit will require a reboot, so it may be easier to set the Advanced Parameter for the PE queue to a larger value before considering changing the HBA value, therefore if it is necessary to increase the HBA queue, the PE will already be set. Note that the value of Scsi.ScsivVolPESNRO will not validate against the HBA value upon reboot.

In general, Dell EMC does not recommend making any changes to the queues, even though there is a single Protocol Endpoint. Testing has shown the defaults are adequate for the vast majority of environments. If it is necessary, however, follow the previous instruction.

vVols with Oracle Database 12c – a practical example

This paper makes every effort to help the user understand and work with virtual volumes. While the walkthroughs explain a step-by-step procedure, however, they do not provide a larger, concrete example of vVols in the real-world. This section aims to show the user how an application, an Oracle 12c database, can be deployed on a virtual machine with virtual volumes rather than VMFS or RDMs. This example uses the initial release of the VASA Provider and vSphere 6.0.

Oracle Database 12c

Oracle database 12c is Oracle's latest release of their relational database management system (RDBMS). What distinguishes an RDBMS is that it moves data into a database, stores it for future retrieval, and then when requested retrieves that data. An RDBMS recognizes two types of operations – logical and physical. A logical operation is one conducted at the application level – for example a request for information from a table. A physical operation is one at the database level, where the database decides how to service that request and return the data.

An Oracle 12c database can be created on a traditional filesystem (e.g. NFS, Windows NTFS drives) or on Oracle's proprietary volume manager and file system, known as Oracle Automatic Storage Management (ASM). When using ASM, Oracle controls where the files live and manages them for the user. ASM groups disks together into disk groups where the data is stored. In order to use ASM, the Oracle Grid Infrastructure is installed and configured. ASM runs as an "instance" on the host which is a combination of memory structures and processes. ASM can be used in a single instance (one host) mode or also as the foundation for Oracle Real Application Clusters. For this example ASM will be utilized.

Oracle on Virtual Volumes

Creating an Oracle database on virtual volumes with Oracle is most akin to using VMFS datastores. In both cases, individual disks (vmdks) are added to a VM and created in a datastore. With vVols this happens to be a vVol datastore as opposed to VMFS. Note, however, if one were to browse each type of datastore, VMFS in Figure 144 and vVol in Figure 145, the contents of a VM are the same – virtual disk files, VM log files, memory file, etc. From this view it is not possible to tell the difference (naming aside).

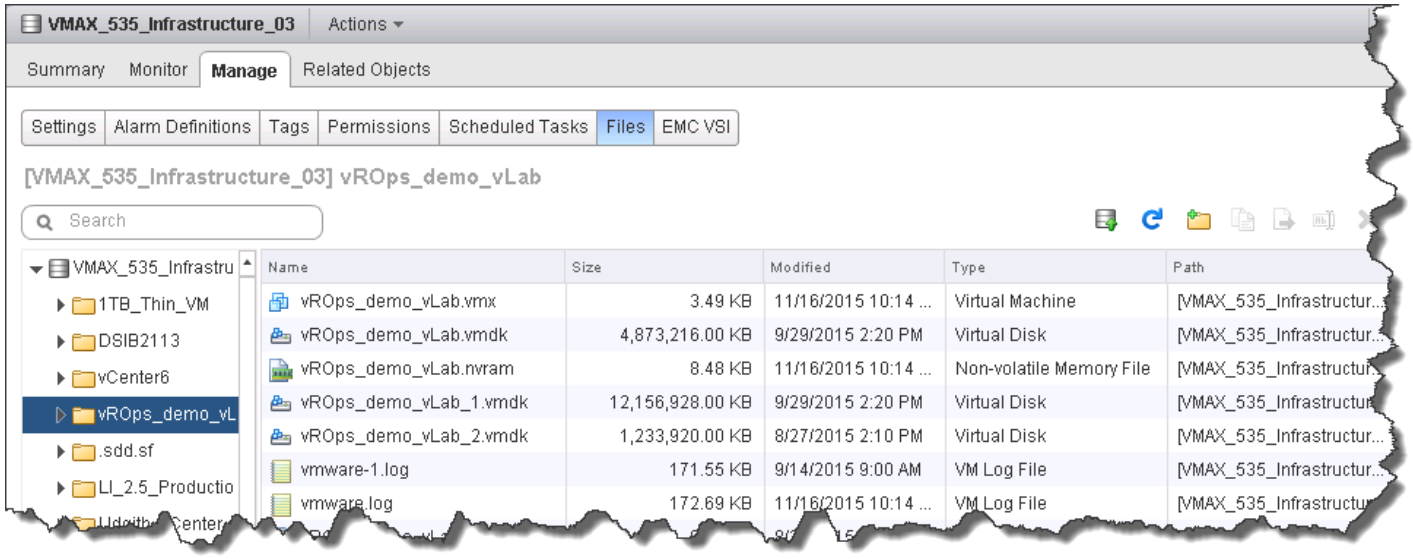

Figure 144. VMFS datastore VM contents

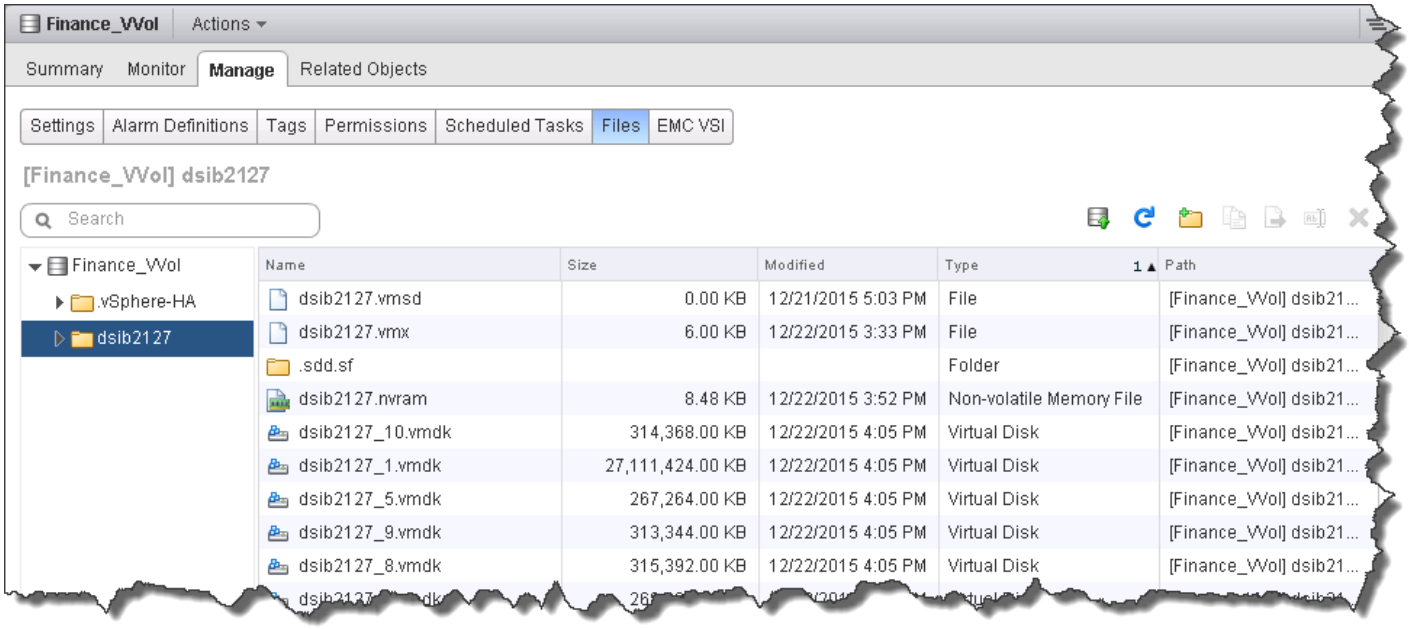

Figure 145. vVol datastore VM contents

The process of creating the VM is also essentially the same, save that the user selects a vVol datastore utilizing the VM Storage Policy which ensures the proper SLO and workload is selected. When creating a VM that will require many disks, as in the case of an Oracle database, each subsequent Hard disk added will default to the VM Storage Policy initially created. For instance, in Figure 146, during VM creation, the Finance_Diamond VM Storage Policy is selected and then the compatible vVol datastore Finance_vVol.

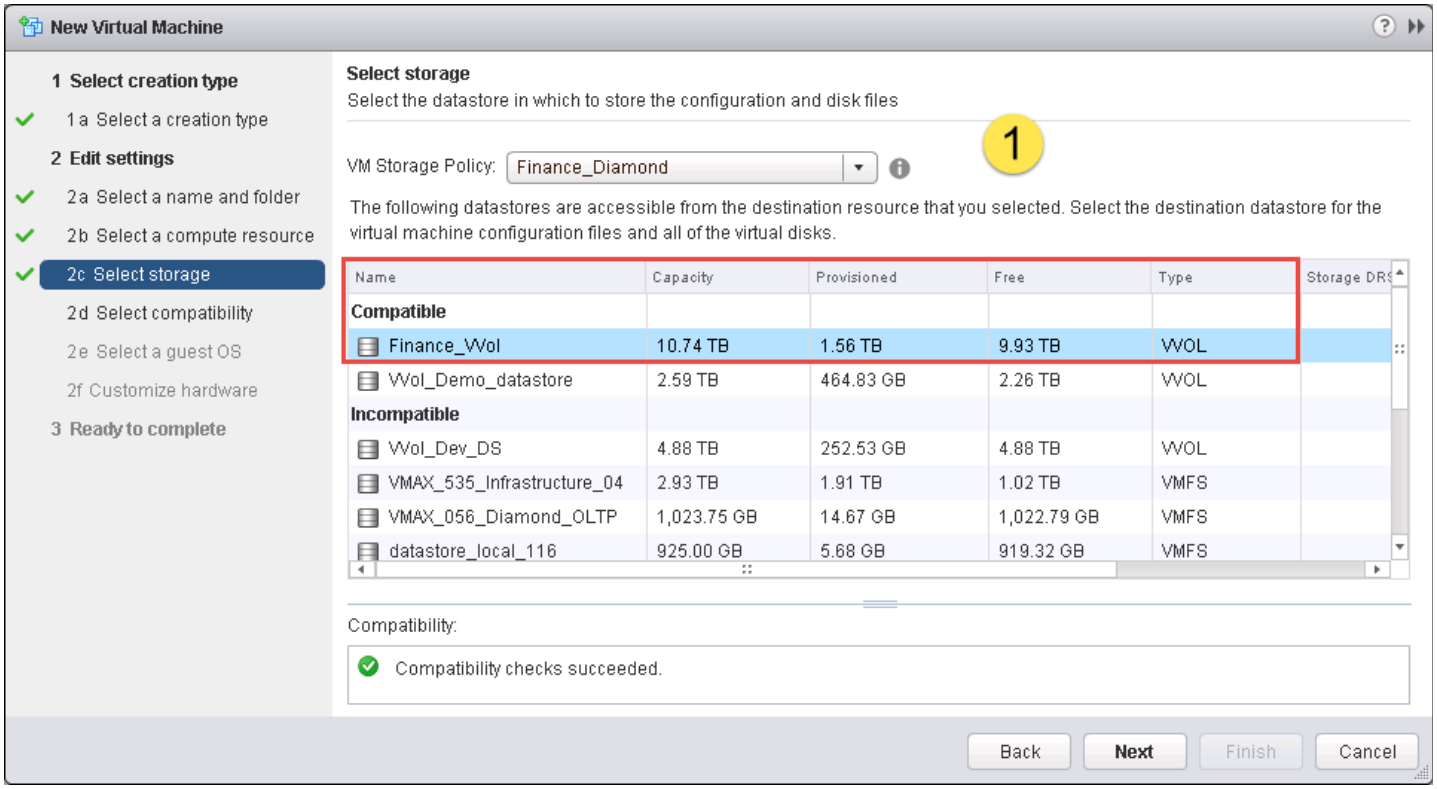

Figure 146. VM Storage Policy selection for datastore

When the Customize hardware step (2f) is reached and a new disk is added as in Figure 147, the VM storage policy automatically defaults to the original policy chosen. Each disk can be added in this manner if desired. If Hard disks are added after VM creation, the storage policy still defaults to the original policy, even if the last disk added to the VM was assigned a different policy.

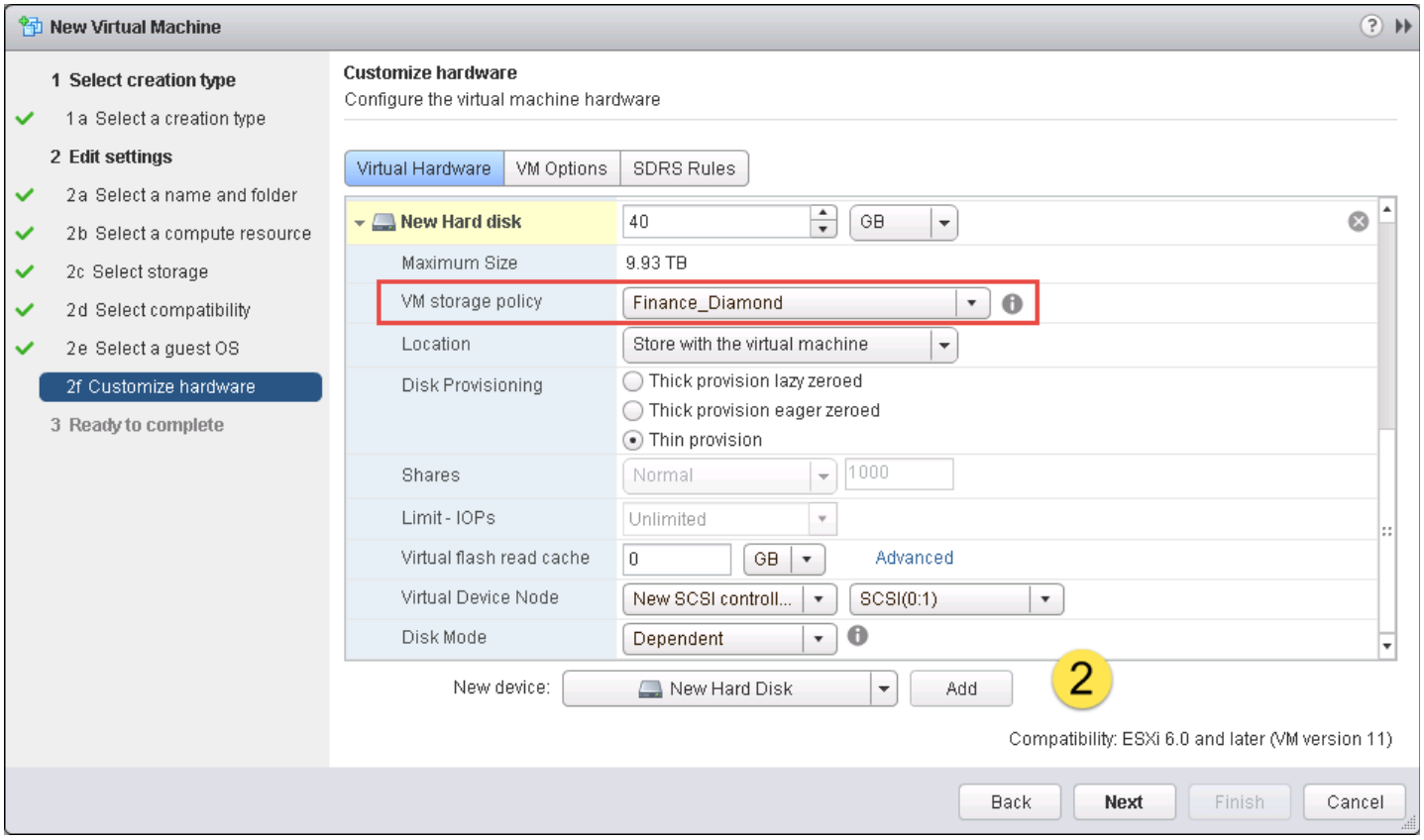

Figure 147. VM Storage Policy selection for individual Hard Disk

In the case of deploying an Oracle database with ASM, it may be desirable to use different SLOs for one or more ASM diskgroups. This is easily accomplished by adjusting the VM storage policy for each Hard disk to match the desired SLO. For the environment covered in this example, the Oracle database required by the application is relatively small and requires the best possible performance. Therefore the entire database will be placed in a Diamond+OLTP SLO.[12]

Environment

As mentioned, the Oracle database is a single instance on the ASM file system. The VM is configured with 18 disks, 1 for the Oracle Enterprise Linux OS 5, 1 for the Oracle 12.1.0.1 binaries, and 16 for the Oracle database. The VM details are present in Figure 148.

[12] For Oracle database best practices on the VMAX, please see the following whitepaper Deployment Best Practice for Oracle Database with VMAX3 Service Level Objective Management

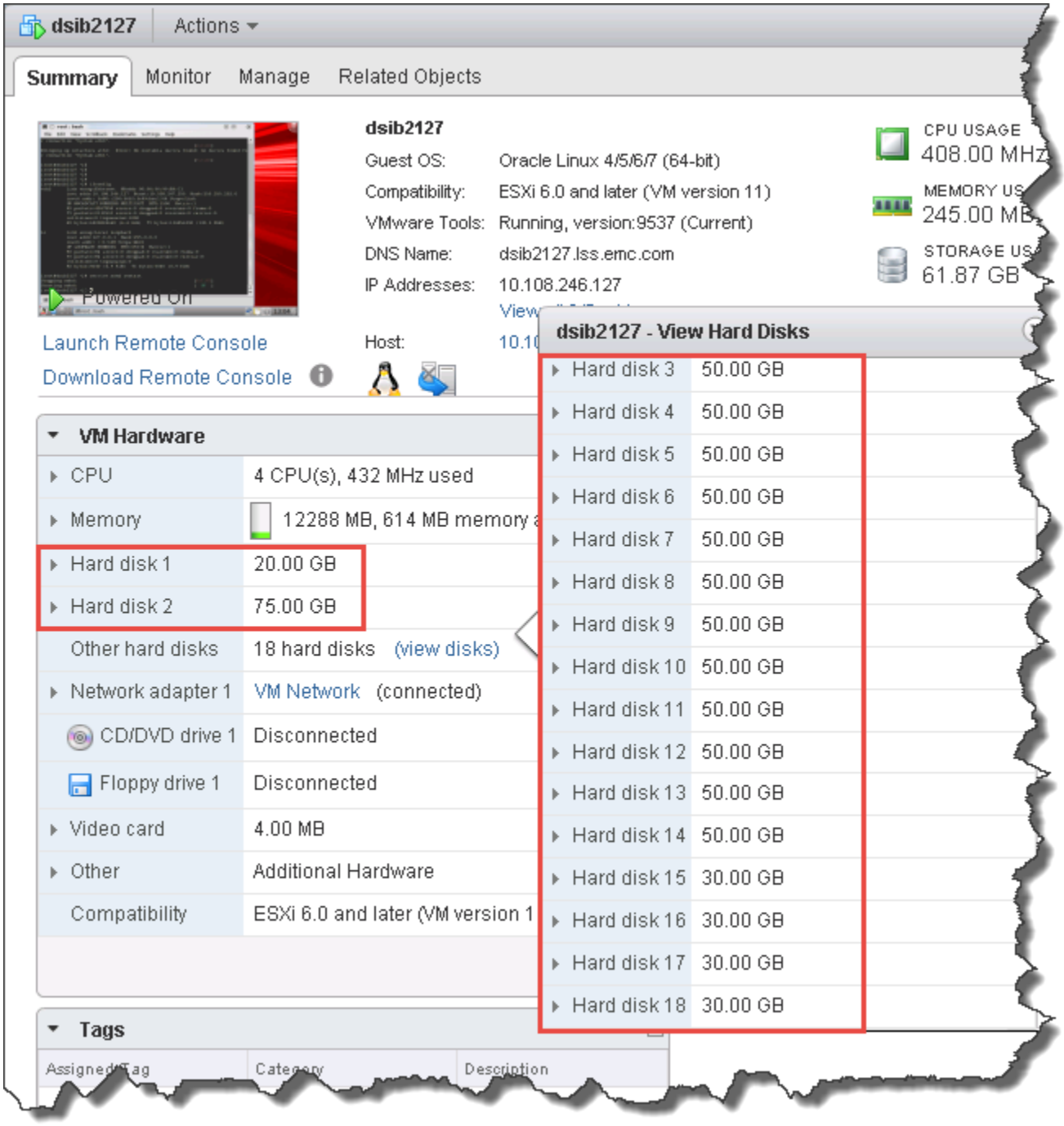

Figure 148. Oracle database VM details

The details of the ASM setup are present in Table 3.

Table 3. ASM configuration

ASM diskgroup	# of disks	Size of disks	Redundancy
DATA	6	50	External
FRA	6	50	External
REDO	4	30	Normal
Storage Array Detail			
Dell EMC VMAX	HK000197200056	VMAX 400K	5977.691.684 microcode

It is important to remember that as far as the operating system is concerned, a vVol vmdk is no different than a VMFS vmdk. The operating system simply sees disks. So once the VM is configured, the process for installing the Oracle binaries and creating the database is the same.

All disks should be configured with fdisk. Create a single primary partition and using expert mode the offset should be set to 128. This will align the disks to the VMAX storage since individual vmdks are not aligned by default. The one disk that is slated for Oracle binaries can have a regular file system installed (ext4). The other disks that are designated for ASM will be used in this environment with the ASM library. The ASMLib is not required but it does make configuration much easier. Once oracleasm is initialized and configured with the proper user and group, the disks can be labeled for use with ASM. The syntax to create an ASM disk is:

oracleasm createdisk DATA_1 /dev/sdc1

When you use the ASMLib be sure to change the ASM discovery path during ASM instance create to: /dev/oracleasm/disks.

Availability

As previously mentioned in the section Replication, VMware does not currently support replication in VASA 2.0. It is possible, however, to enable high availability on the VM within an HA VMware cluster. As long as each host in an HA cluster has a Protocol Endpoint and has the vVol datastore mounted (see vVol datastores in a cluster), the VM will be protected with HA as is the case with the Oracle VM in Figure 149. Recall, however, that while HA is supported, VMCP is not.

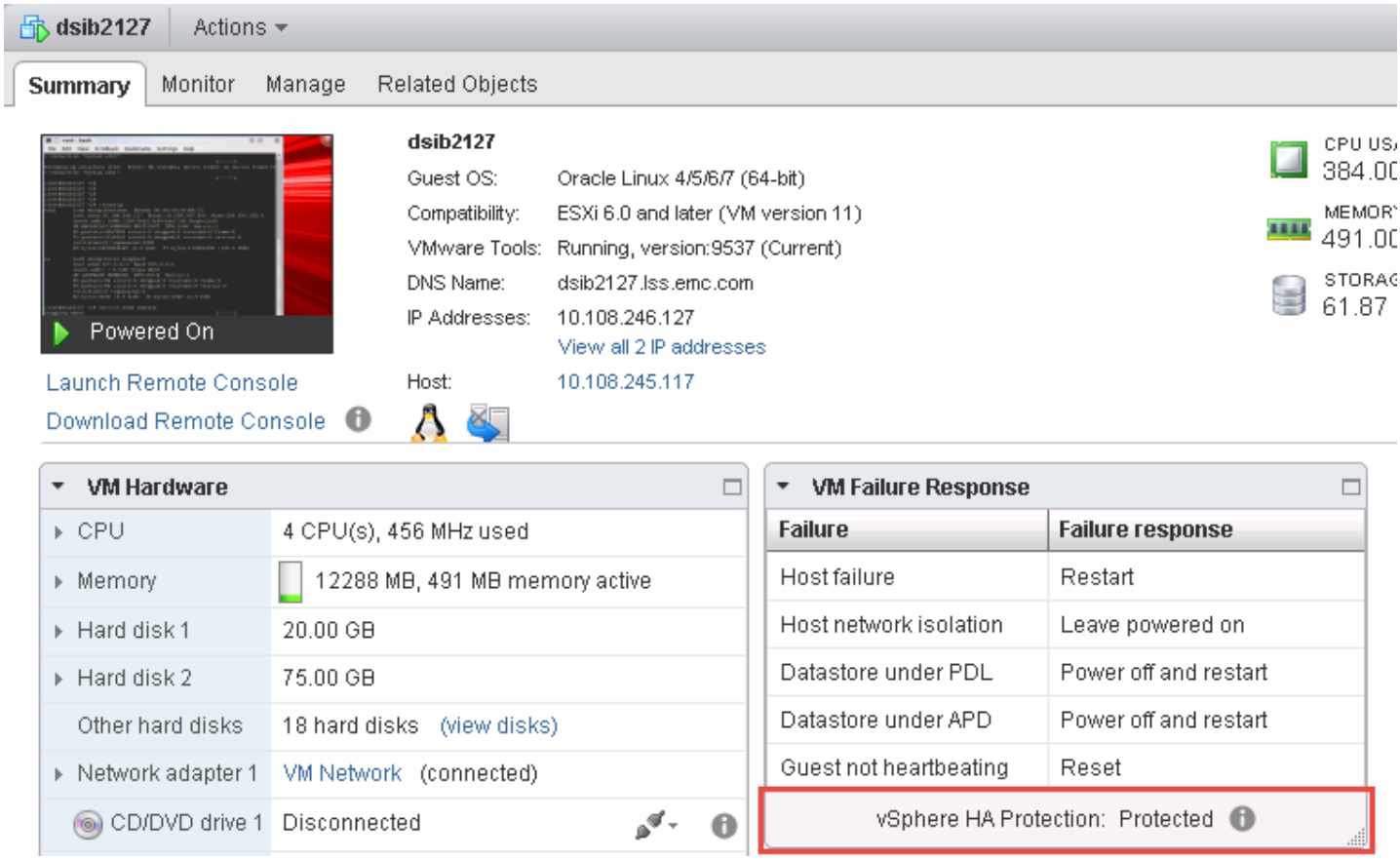

Figure 149. HA with vVols

For disaster recovery of Oracle databases, until VMware supports replication, there are many software solutions available that can be utilized such as Oracle Data Guard.

Snapshots

One of the benefits of using virtual volumes over RDMs is that the VM can be snapshotted (VMware initiated). Taking a snapshot provides a point-in-time copy of the Oracle database. With vVols no longer are snapshots comprised of VMware delta files which occupy more space in the datastore and can grow bigger than the original VM. VMware now passes off the snapshot to the VMAX through the VASA Provider. This allows the VMAX to use TimeFinder/SnapVX technology to take the snapshot. The snapshot will not create any extra devices, save for a memory vVol if chosen. This is tremendously beneficial in comparison to VMFS technology since no additional storage is taken and VMware does not need to keep track of changes after the snapshot, saving precious resources. It is possible to take up to 12 snapshots of a VM.

Currently there are some limitations with how vVol devices can be manipulated. For example, snapshots of vVols cannot be taken outside of the VMware interface. It is not possible to use Unisphere or Solutions Enabler to make copies of vVols that comprise a VM.

Backups

In order to recover from an Oracle database backup and roll forward to the most current archived transaction[13], it is necessary to have copies of the archived redo logs. As mentioned, there are different methodologies that one might employ to maintain a remote copy of the database, but this example is designed to show specifically how to use a copy of a vVol Oracle database to recover.

In order to ensure that archive logs are available for recovery at a secondary location, it is essential they are stored in multiple destinations. Oracle natively permits sending multiple copies of archive logs to multiple directories. Since the VMAX not only supports vVols but also eNAS, using an NFS directory is a good solution to the problem. There are two possibilities one might choose from when using NFS as the secondary archive location in this example. Once the file system is created and exported on the VMAX, it could be mounted directly to the Oracle Linux OS as a shared mount point. Any backup server could also use the same mount point and thus have access to the archive logs. The other option is to create an NFS datastore in vSphere which would also serve the same purpose. In this case, it is a better option to use an NFS mount. Assuming read/write access to all, an NFS mount's contents can be seen and manipulated equally by all participants. This means that both a production VM and a backup VM can see the archive logs on that mount. If an NFS datastore is utilized, a new Hard disk will be needed on the production VM and then it would need to be shared with another VM. It unnecessarily complicates the solution, particularly when using an NFS mount consists of two commands – creating the directory and mounting it as in Figure 150. Note that an SLO of Diamond was selected, just as with the rest of the Oracle database.

[13] If an Oracle database crashes, it is likely that transactions still exist in the redo logs that have not been archived. The assumption here is that the redo logs are no longer accessible so the user only has access to the archive logs.

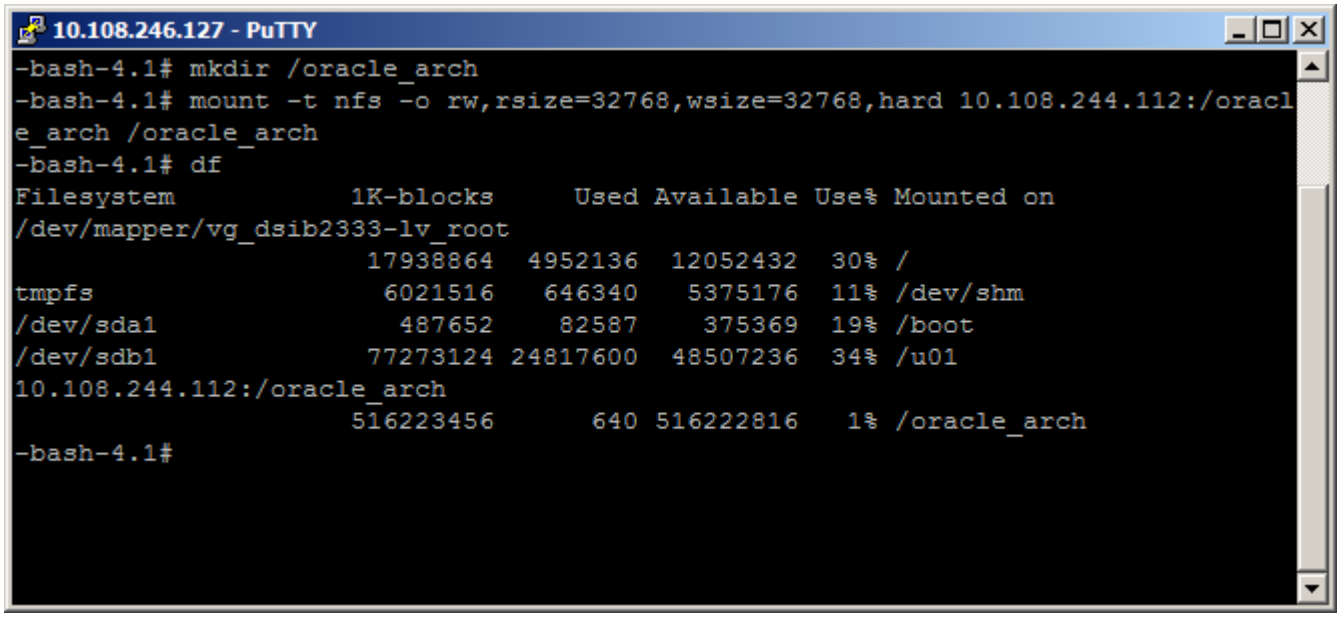

Figure 150. NFS mount for Oracle archive logs

Once created be sure to modify the pfile/spfile for the database to add the secondary archive location and that the NFS location has read/write privileges for the oracle owner. If necessary, restart the database. Figure 151 shows the resulting changes with the first archive destination on ASM, and the second one on the NFS mount.

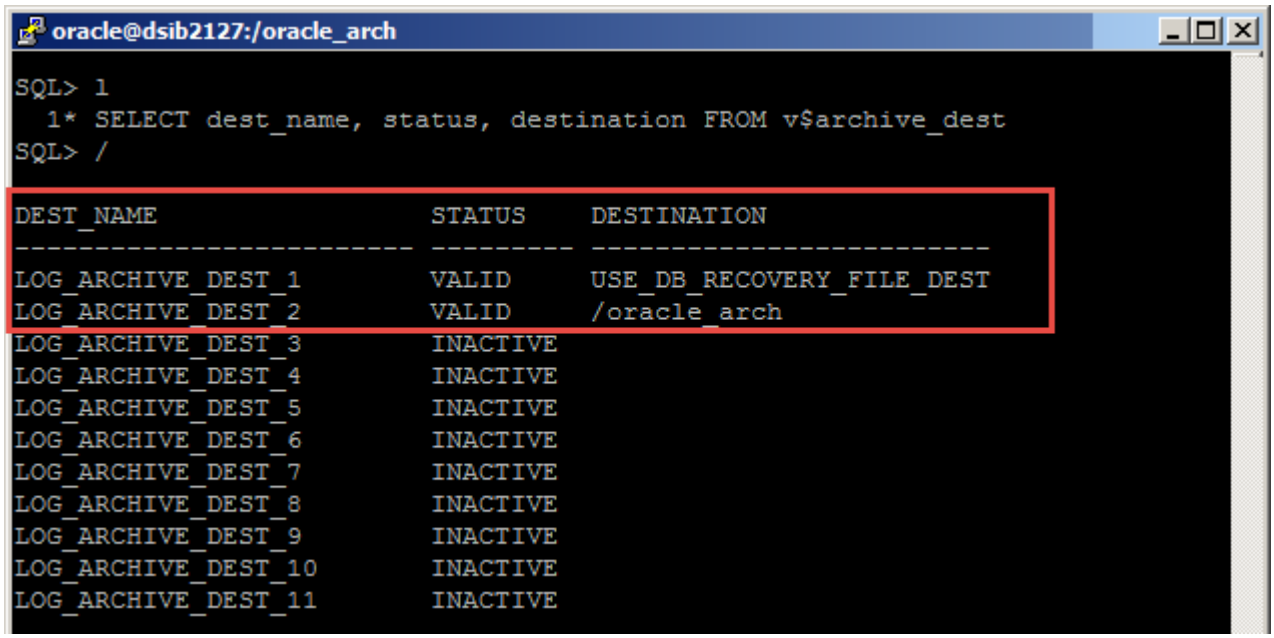

Figure 151. Duplex archive log destinations

Oracle database backups can be taken directly from the production VM or from a backup VM. As mentioned, since vVols cannot be acted upon directly by TimeFinder commands (in order to take a consistent copy), the best way to make the backup server is through a VM clone.

There are ways of using existing snapshots of the production VM as the source of a clone, but they are script-based and more complicated, yet produce the same result as a regular clone.

If using a VM clone be sure to place the database in backup mode before cloning the VM to ensure the database can undergo recovery.[14] At a high level the following steps were taken:

- On production database issue: *alter database begin backup;*
- Clone the production VM and use the customization option of the clone to change the IP and hostname
- Once clone completes, on production database issue: *alter database end backup;*
- On clone VM:
  - Mount the NFS share
  - Run as root from $GRID_HOME/crs/install: *perl roothas.pl -deconfig –force*
  - Run as root from $GRID_HOME: *root.sh*
  - Run as root: *srvctl add asm*
  - Run as root: *srvctl start asm*
  - Start database in mount only mode
  - Backup with RMAN

For a detailed description of backup and recovery procedures of Oracle on the VMAX (not vVol specific) see the engineering whitepaper: ORACLE DATABASE BACKUP, RECOVERY, AND REPLICATIONS BEST PRACTICES WITH VMAX ALL FLASH STORAGE.

Conclusion

VMware's storage paradigm, vVol, brings storage management from the VMFS datastore level, down to the virtual machine. Virtual volumes on the VMAX are individual TDEVs which are mapped directly to a vmdk on a VM. Such granularity permits the VMware administrator the ability to customize a VM according to the services offered by the VMAX – SLO and workload (if applicable) as well as compression and deduplication (if applicable).

Managing storage tiers, provisioning, migrating, cloning virtual machines and correct virtual machine placement in vSphere deployments have become more efficient and user friendly with VASA 2.0 and vVols. It removes the need for maintaining complex and tedious spreadsheets and validating compliance manually during every migration or creation of a virtual machine or virtual disk.

[14] It is possible to recover an Oracle database without hot backup mode.

This white paper discussed how to install, configure and use Virtual Volumes in VMware vSphere 6 and 7 environments with VMAX and PowerMax storage arrays. An understanding of the principles that were exposed here should allow the reader to deploy and utilize VMware vSphere in the most effective manner with vVols.

References

Dell EMC

- *Using Dell EMC VMAX and PowerMax Storage in VMware vSphere Environments* TechBook
  http://www.emc.com/collateral/hardware/solution-overview/h2529-vmware-esx-svr-w-symmetrix-wp-ldv.pdf
- *Using VMware Storage APIs for Array Integration with Dell EMC VMAX and PowerMax*

  http://www.emc.com/collateral/hardware/white-papers/h8115-vmware-vstorage-vmax-wp.pdf
- *Unisphere for VMAX*

  https://support.emc.com/products/27045_Unisphere-for-VMAX/Documentation/
- *Unisphere for PowerMax*

  https://support.emc.com/products/44740_Unisphere-for-PowerMax/Documentation/

VMware

- *vSphere Documentation*

  https://docs.vmware.com/

Appendix: VASA Provider/Virtual Volume Troubleshooting

DNS

Perhaps the most important component of networking in the VMAX vVol solution is DNS. Failure to properly configure DNS can cause many different communication issues between the VASA Provider, ESXi and the vCenter.

Starting with the VASA Provider, if the proper DNS is not configured and available when the vApp is deployed, the host name will not be resolved from the IP, and the vApp will be named localhost.localdomain. The naming will not prevent the user from proceeding through the initial steps of mapping Gatekeepers and the VASA DB in the section Installation, however after mapping the devices, they will not appear in the bottom window as in Figure 152.

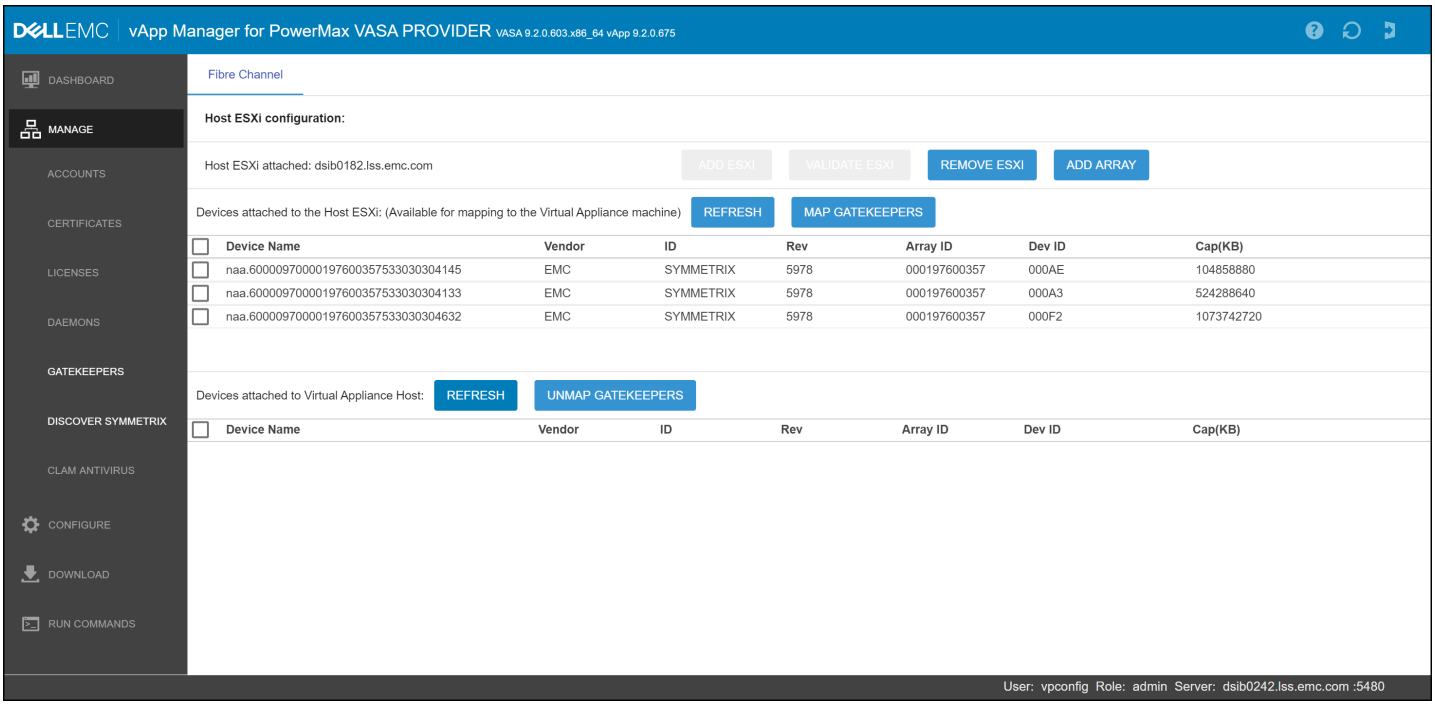

Figure 152. Mapped GATEKEEPERS do not show in bottom panel

If lack of a host name occurs because of a failure to use the proper DNS, then the easiest solution is to power off the VP, delete it, and make sure the IP resolves the proper hostname from DNS before redeploying. For instance, in this paper the VP hostname is dsib2122.lss.emc.com with an IP of 10.228.246.122. Before deploying, both the IP and hostname were resolved properly as in Figure 153.

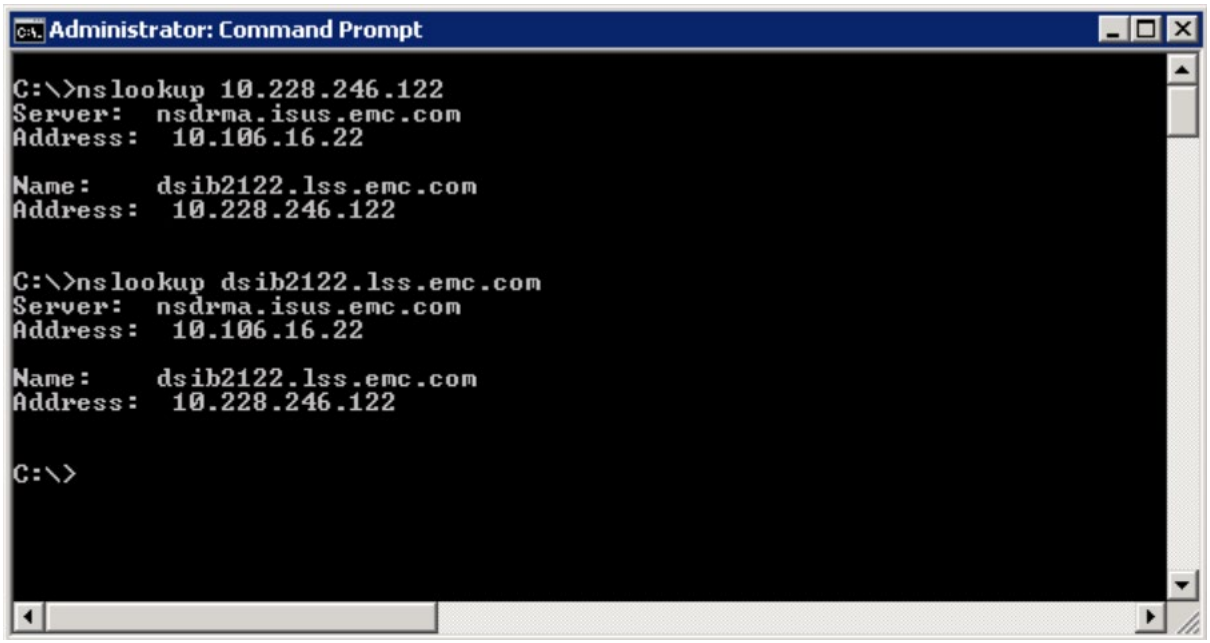

Figure 153. DNS resolution for VP

If, however, the proper DNS was used and the name resolution works, the easiest solution is to rename the host. Navigate to CONFIGURE -> HOST and use the Change Host Name function in Figure 154. The vApp should reboot after setting the new host name. If it does not, be sure to manually reboot.

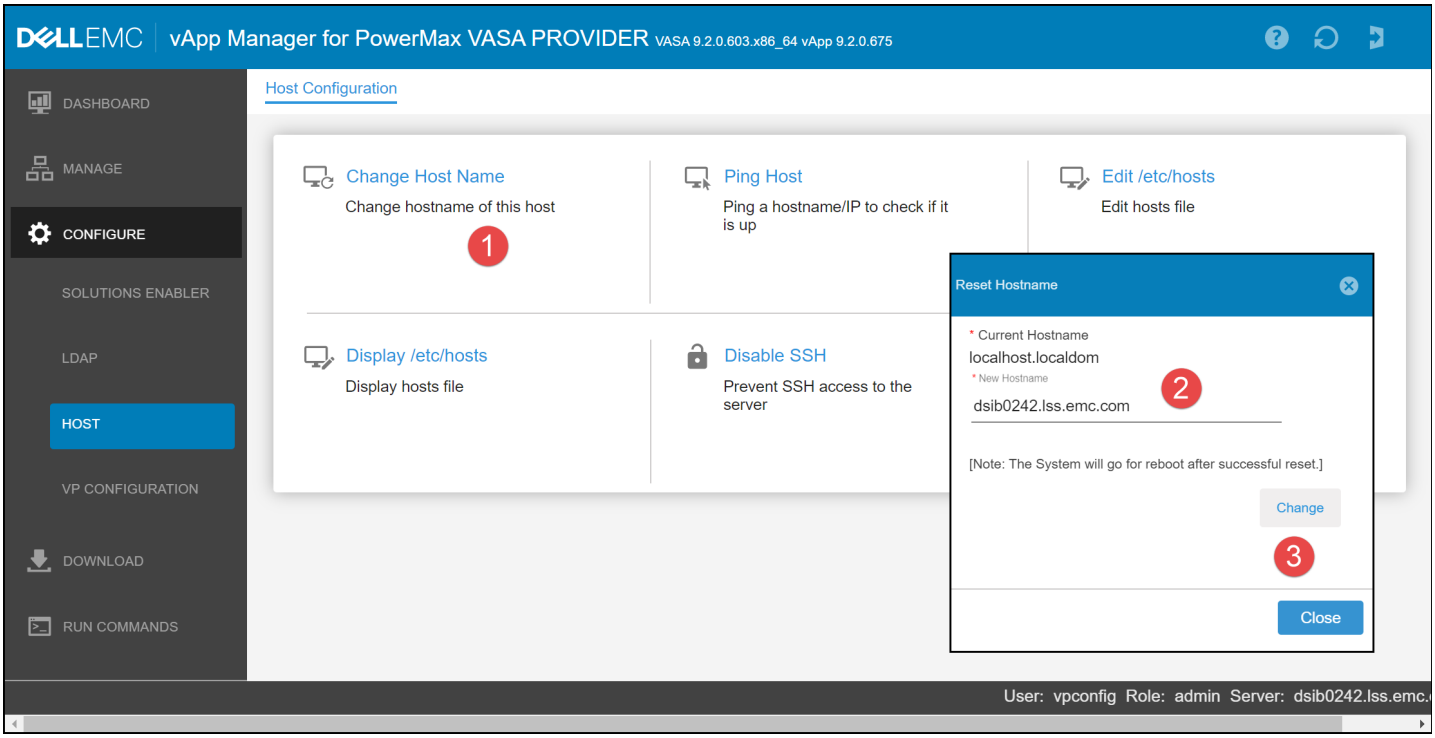

Figure 154. Rename host in VP

Upon reboot, the Gatekeepers are now visible in Figure 155.

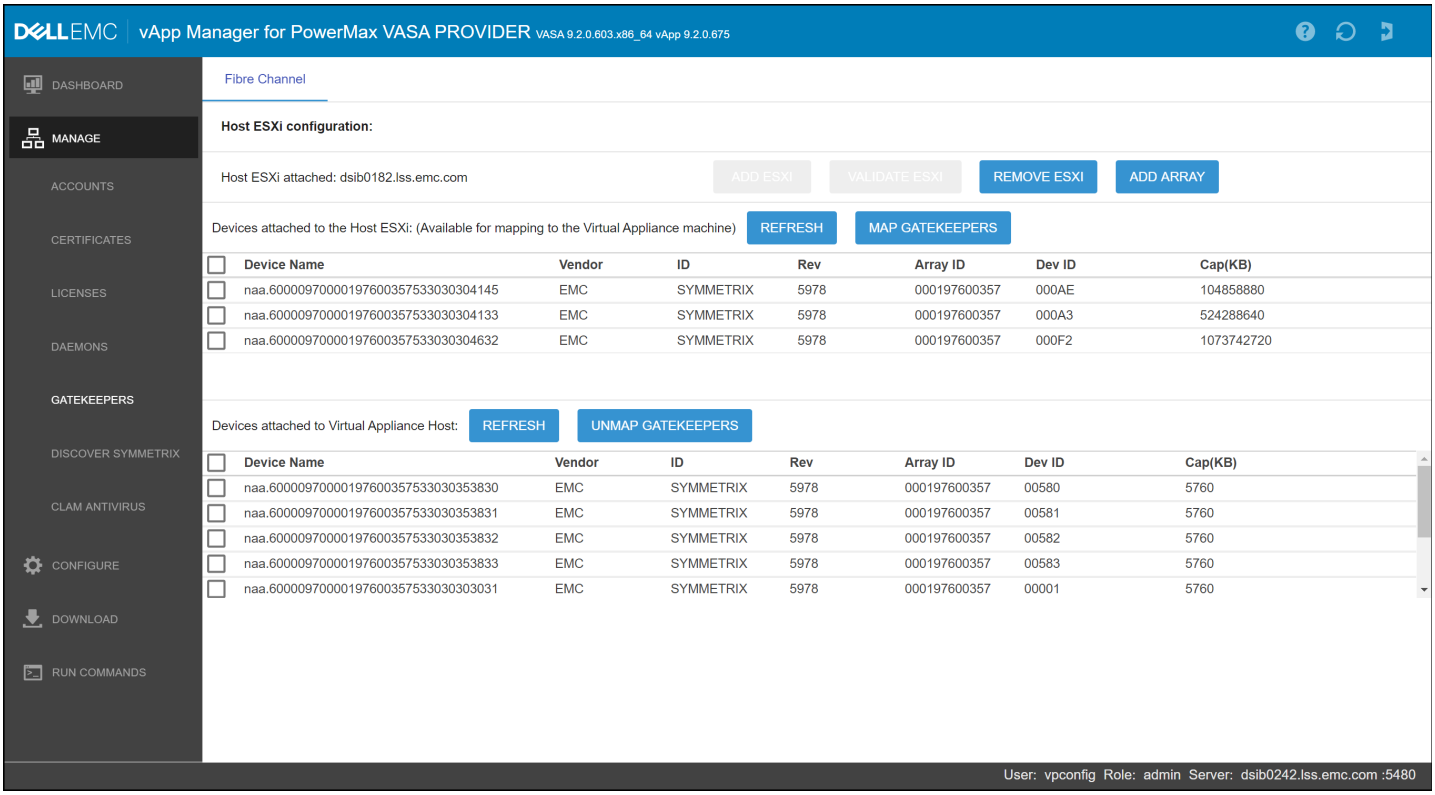

Figure 155. Gatekeepers now visible

Additionally, if the VP is not configured correctly with the DNS, registration in the vCenter will fail.

ESXi host DNS

On the ESXi host, if DNS is improperly or not configured at all, when the vVol datastores are created their size will show as zero bytes (Figure 156). In addition, after some time the datastore will show as inaccessible and the PE will not display.

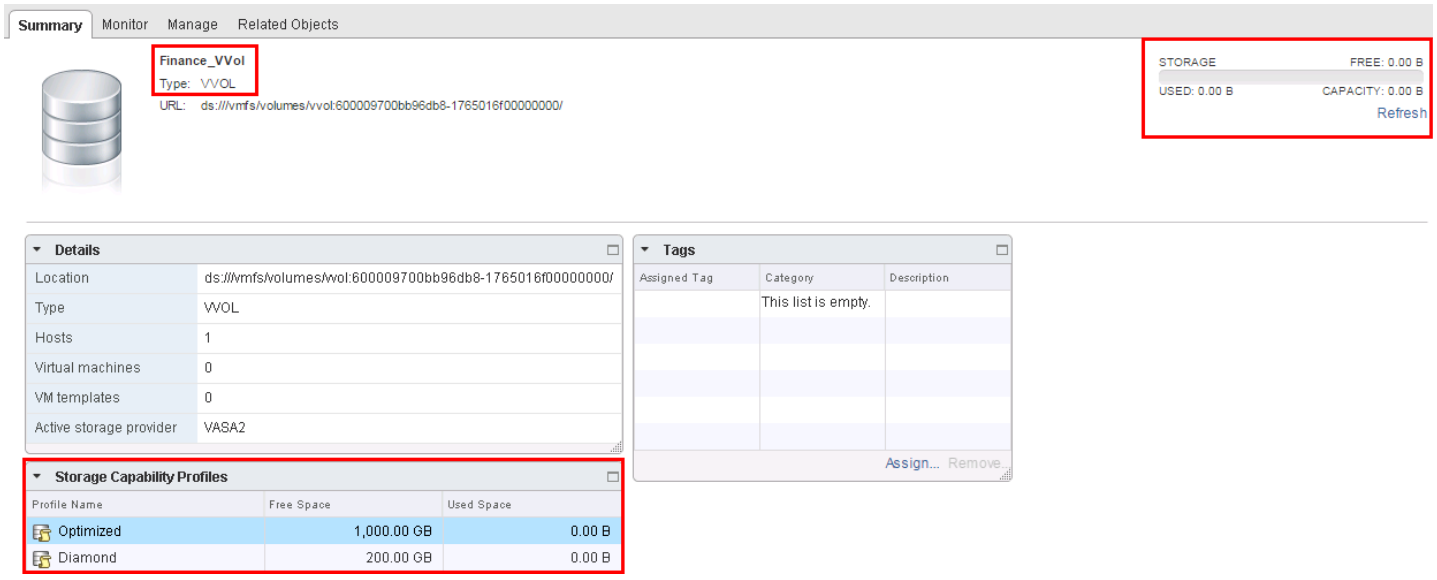

Figure 156. vVol datastore with zero bytes

Once the DNS is fixed on the ESXi host, the vVol datastores will return the correct size.

Validate ESXi host

When validating the ESXi host in Figure 13, there is a known issue in vSphere 6.7 and higher where the vApp will report that it “Failed to retrieve the issuer certificate” seen in Figure 157.

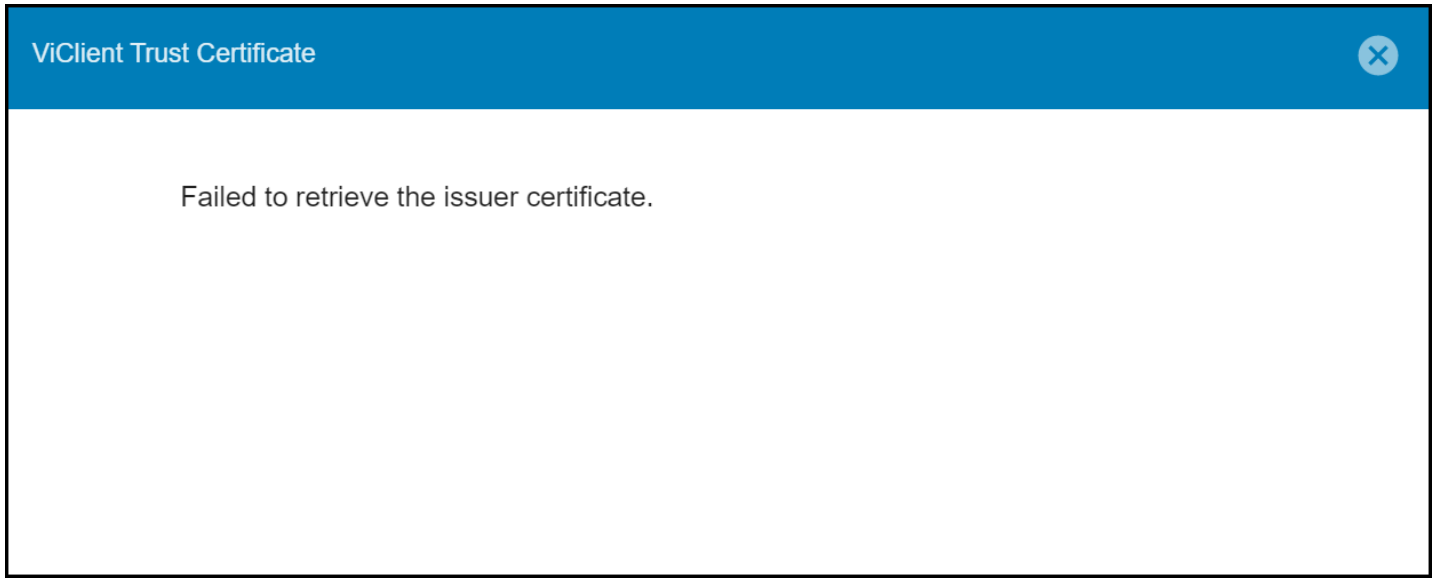

Figure 157. Validate ESXi host error

To fix this, a change is required to the /etc/vmware/rhttpproxy/config.xml file on the ESXi host that is trying to be validated. Specifically, remove the comment-out characters from this line:

Edit the file and change this:

‹!-- ‹keyStoreFile›/etc/vmware/ssl/castore.pem‹/keyStoreFile› →

To this and save:

‹keyStoreFile›/etc/vmware/ssl/castore.pem‹/keyStoreFile›

After making the adjustment, run the command "/etc/init.d/rhttpproxy restart". After the service restart completes, re-attempt validation. This issue is covered in VMware KB article 60216.

Certificate Issues

There is a second scenario where the datastore will show zero bytes and the PEs will not be visible, and that is if the ESXi host is unable to register with the VASA Provider due to a certificate mismatch. In this scenario the user will be able to register the VASA Provider in the vCenter successfully and create the datastore, however the datastore will be zero bytes and the PE will not display in the screen as it does in Figure 158.

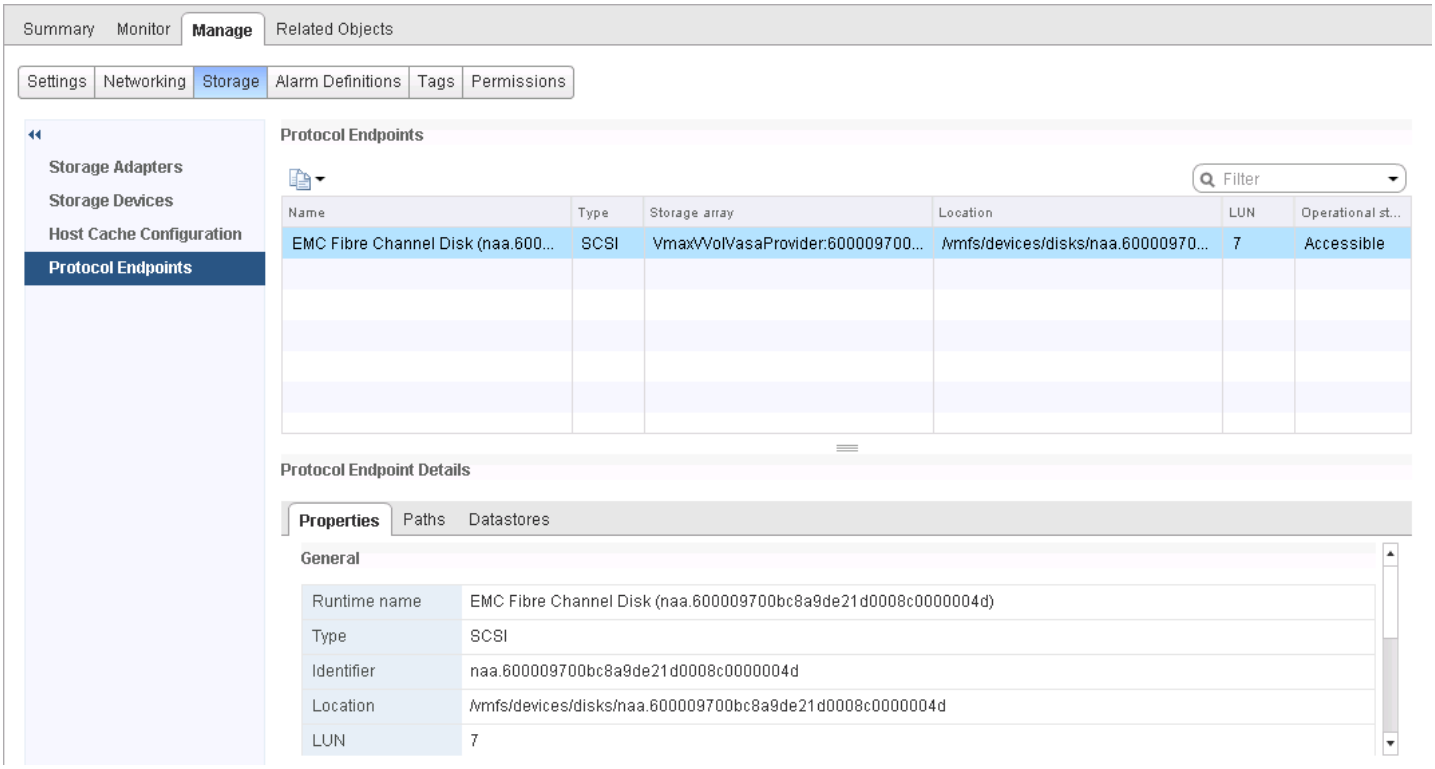

Figure 158. Protocol Endpoint screen

A certificate problem will be apparent in the cimomlog.txt file with entries similar to this one:

ECOM WebServer: Net Exception occured [sic] while doing SSL handshake. Error accepting connection.

Unfortunately, the error message is not limited to this particular problem so it is the combination of symptoms which may indicate a certificate issue.

In order to resolve the mismatch, the user should refresh the CA certificates on each ESXi host. Before doing so the user should unmount the existing vVol datastore and unregister the VASA Provider. Once that is complete, to refresh the certificates, follow the menu in Figure 159.

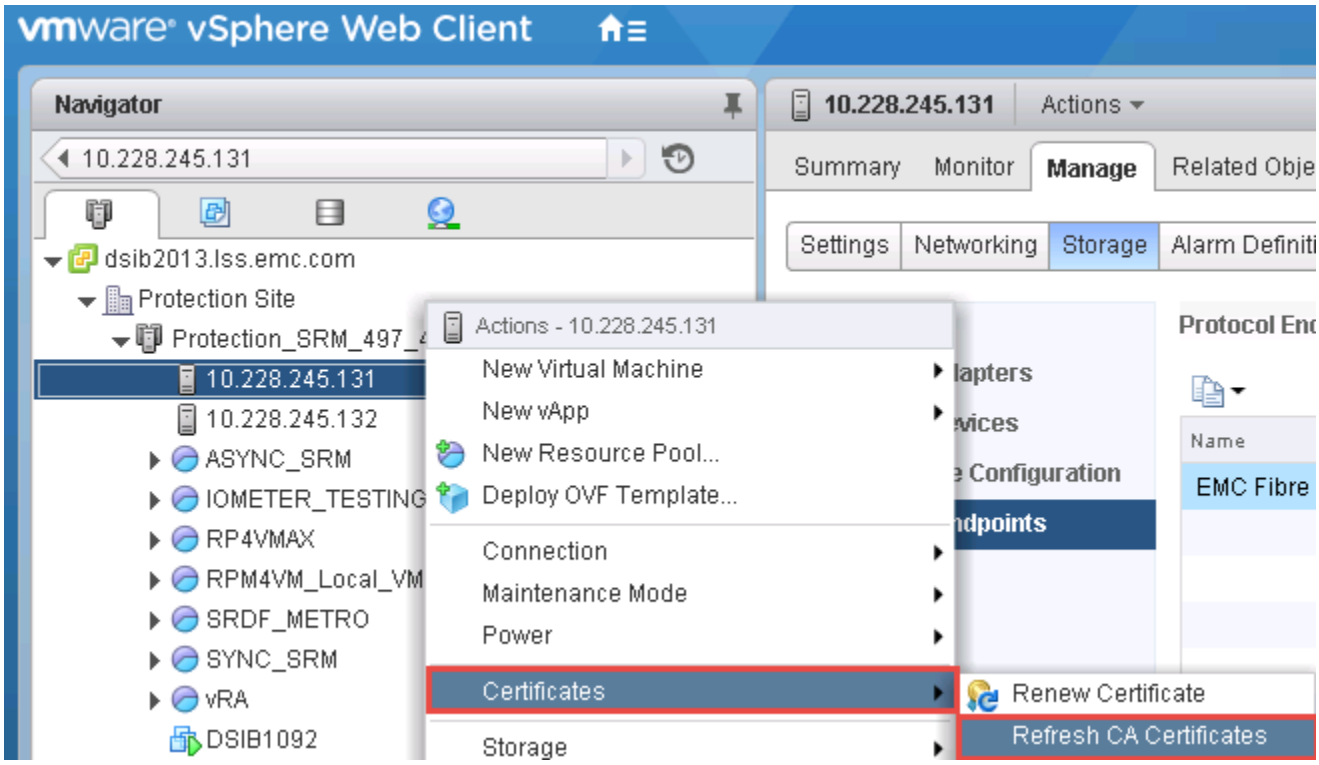

Figure 159. Refresh CA certificates

Once the certificates are refreshed, all management services need to be restarted on each ESXi host. This is a disruptive action. Execute the following as root:

services.sh restart

When everything is restarted, re-register the VASA Provider and re-create the datastore. A non-zero datastore will indicate success.

Cascaded Groups/Multi-host initiator groups

As noted in the paper, currently there is no support for cascaded auto-provisioning groups for the Protocol Endpoint. If the PE is part of a masking view with cascaded objects, there will be various issues with the implementation on the vCenter. Neither Unisphere for VMAX nor Solutions Enabler will prevent the user from provisioning the

PE with cascaded groups so it is important to check this first if issues are encountered.

In addition to the impermissibility of cascaded initiator groups, an initiator group used in a PE masking view may not contain initiators from multiple hosts. Each ESXi host in a cluster (for example) must have a unique PE. To avoid any issue, it is advisable to have a separate initiator group for each ESXi host. Note, however, it is still possible to have a parent initiator group for use in non-PE masking views.

VASA Provider Recovery

If the VASA Provider virtual appliance fails or is lost, it is possible to redeploy and use the existing VASA DB device in the new vApp. Simply deploy the VP again using the same or even a different IP/hostname, present Gatekeepers from the array, and also present the original device used for the VASA DB. In the GATEKEEPERS screen, mount the original VASA DB device. Figure 18 shows an image of the dialog box which the user is presented with when trying to mount an existing VASA DB. Select 'No' to mount the existing image. After the VASA DB is mounted, it will be necessary to unregister the VP from the vCenter(s) and re-register. This should be done regardless if the same IP/hostname was used in the new VP deployment. If the VP is not re-registered, any rescans of the lost VP will fail regardless if the hostname of the new VP is the same.

If the VASA DB itself is corrupted in some manner or the device destroyed, it can be reconstructed from the array. The reconstruction, however, will require the assistance of Dell EMC Support.

Orphaned Virtual Volumes

While unlikely, it is possible that a vVol fails to be removed when a VM is deleted. It is important to understand that this is not the same as removing a vVol (vmdk) from a VM but choosing not to select the box to delete the file. In that case, it is possible to manually delete the vmdk from the datastore, or to add the vmdk back to a VM. The case of an orphaned vVol means that the vmdk is deleted, but the backing vVol on the array is not. Therefore, no vmdk exists to delete. As a Storage Resource cannot be deleted while a vVol(s) exists, this is the most common way a user might discover an orphaned vVol. As this is an unexpected condition, there is currently no process for a user to remove an orphaned vVol. If such a condition arises, please contact Dell EMC Support who can resolve the issue through an internal process.