Table of contents

Introduction ........................................... 4
  vSphere Virtual Volumes components ................. 6
  Vendor provider or VASA provider .................. 7
  Storage container ................................... 8
  Virtual datastore ................................... 8
  Protocol endpoints ................................ 10
  vSphere Virtual Volumes requirements ............ 11
  Software ........................................... 11
  Hardware ......................................... 11
  License ............................................ 11

Configuring vSphere Virtual Volumes ............... 12
  Configuration guidelines ................................ 12
  Storage ........................................... 12
  vSphere .......................................... 12
  Configuration procedures ............................ 12
  vSphere storage time synchronization ................ 12
  vCenter Server time synchronization .............. 12
    ESXi host time synchronization .................. 12
    vSphere host time synchronization configuration procedure 13
    Storage provider registration ................... 14
  Storage provider registration procedure ......... 14
  Virtual datastore creation procedure .............. 15
  Mounting the virtual datastore onto multiple hosts .. 18
  Mapping storage capabilities to VM storage policies .... 19
  Mapping storage capabilities to VM storage policies procedure 19
  Virtual machine creation .......................... 22
  Virtual machine migration with vSphere Storage vMotion .... 24
vSphere Virtual Volumes interoperability .................................................. 27
vSphere Enterprise features ........................................................................ 27
VMware products and solutions .................................................................. 27
vSphere Virtual Volumes CLI commands .................................................... 28
Virtual Volumes ESXCLI namespaces ....................................................... 28
ESXCLI storage Virtual Volume command line syntax samples ................ 28
Virtual Volumes ESXCLI namespace commands ........................................ 28
ESXCLI storage Virtual Volume storagecontainer command line syntax samples ................................................................. 28
ESXCLI storage Virtual Volume storagecontainer abandonedvvol syntax samples ................................................................. 29
ESXCLI storage Virtual Volume storage container list syntax sample .......... 29
ESXCLI storage Virtual Volume storagecontainer daemon unbindall syntax sample ........................................................................ 30
Reference .................................................................................................. 32
vSphere Virtual Volumes resources ............................................................ 32
Product documentation ................................................................................ 32
Documentation .......................................................................................... 32
Presentations .............................................................................................. 32
About the authors ....................................................................................... 32
Introduction

VMware vSphere® Virtual Volumes™ implements the core tenants of the VMware software-defined storage vision to enable a fundamentally more efficient operational model for external storage in virtualized environments, centering it on the application instead of the physical infrastructure.

vSphere Virtual Volumes enables application-specific requirements to drive storage provisioning decisions while leveraging the rich set of capabilities provided by existing storage arrays. Some of the primary benefits delivered by Virtual Volumes focus on operational efficiencies and flexible consumption models:

- Virtual Volumes simplifies storage operations by automating manual tasks and reducing operational dependencies between the vSphere admin and the storage admin. By using policy-driven automation as the operations model, provisioning and change management are simplified and expeditated.
- Virtual Volumes simplifies the delivery of storage service levels to applications by providing administrators with finer control of storage resources and data services at the virtual machine (VM) level that can be dynamically adjusted in real time.
- Virtual Volumes improves resource utilization by enabling more flexible consumption of storage resources when needed and with greater granularity. The precise consumption of storage resources eliminates overprovisioning. The virtual datastore defines capacity boundaries, access logic, and exposes a set of data services accessible to the VMs provisioned in the pool.
- Virtual datastores are purely logical constructs that can be configured on the fly, when needed, without disruption and don’t require formatting with a file system.

FIGURE 1: Operational model transformation.
Historically, vSphere storage management has been based on constructs defined by the storage array: LUNs and filesystems. A storage administrator would configure array resources to present large, homogenous storage pools that would then be consumed by a vSphere administrator.

Because a single, homogeneous storage pool would potentially contain many different applications and VMs, this approach resulted in needless complexity and inefficiency. vSphere administrators could not easily specify requirements on a per-VM basis.

Changing service levels for a given application usually meant relocating the application to a different storage pool. Storage administrators had to forecast well in advance what storage services might be needed, usually resulting in the overprovisioning of resources.

With vSphere Virtual Volumes, this approach is fundamentally changed. vSphere administrators use policies to communicate application requirements to the storage array. The storage array responds with an individual storage container that precisely maps to application requirements and boundaries.

![Image of vSphere Virtual Volumes operational model]

Typically, the virtual datastore is the lowest granular level at which data management occurs from a storage perspective. However, a single virtual datastore contains multiple VMs that might have different requirements. With the traditional approach, differentiation on a per-VM level is difficult. The Virtual Volumes functionalities allow for the differentiation of VM services on a per-application level by offering a new approach to storage management.

Rather than arranging storage around features of a storage system, Virtual Volumes arranges storage around the needs of individual VMs, making storage VM centric. Virtual Volumes maps virtual disks and their respective components directly to objects on a storage system. This mapping allows vSphere to offload intensive storage operations such as snapshot, cloning, and replication to the storage system. It is important to
familiarize yourself with the concepts relevant to Virtual Volumes and their functionality. This document provides a summarized description and definitions of the key components of Virtual Volumes.

**vSphere Virtual Volumes components**
The following summarizes the description and definition of the key components of vSphere Virtual Volumes.

Virtual Volumes are a new type of VM object that are created and stored natively on the storage array. Virtual Volumes are stored in storage containers and mapped to VM files/objects such as VM swap, VMDKs, and their derivatives.

There are five different types of Virtual Volumes, each mapping to a different and specific VM file:

- **Config** – VM home, configuration files, logs
- **Data** – Equivalent to a VMDK
- **Memory** – Snapshots
- **Swap** – VM memory swap
- **Other** – vSphere solution-specific object

*FIGURE 3: vSphere Virtual Volumes object types.*
Vendor provider or VASA provider

The vendor provider, also known as the VMware APIs for Storage Awareness (VASA) provider, is a storage-side software component that acts as a storage awareness service for vSphere and mediates out-of-band communication between VMware vCenter Server® and vSphere hosts on one side and a storage system on the other. Storage vendors exclusively develop VASA providers.

vSphere hosts and vCenter Server connect to the VASA provider and obtain information about available storage topology, capabilities, and status.

Subsequently, vCenter Server provides this information to vSphere clients, exposing the capabilities around which the administrator might craft storage policies in storage policy-based management.

VASA providers are typically set up and configured by the vSphere administrator in one of two ways:

• Automatically via the array vendors plug-in
• Manually through vCenter Server

FIGURE 4: Vendor (VASA) provider.
Storage container
Unlike traditional LUN or NFS-based vSphere storage, Virtual Volumes functionality does not require preconfigured volumes on the storage side.

Instead, Virtual Volumes uses a storage container, which is a pool of raw storage capacity and/or an aggregation of storage capabilities that a storage system can provide to Virtual Volumes.

Depending on the storage array implementation, a single array may support multiple storage containers. Storage containers are typically set up and configured by the storage administrator.

Containers are used to define:
• Storage capacity allocations and restrictions
• Storage policy settings based on data service capabilities on a per-VM basis

FIGURE 5: Storage containers.

Virtual datastore
A virtual datastore represents a storage container in a vCenter Server instance and the vSphere Client. A vSphere virtual datastore represents a one-to-one mapping to the storage system’s storage container.

The storage container (or virtual datastore) represents a logical pool where individual Virtual Volume VMDKs are created.

Virtual datastores are typically set up and configured by a vSphere administrator.

FIGURE 6: Virtual datastore.
There is a one-to-one mapping of a Virtual Volume datastore to a storage container on the array. If another Virtual Volume datastore is needed, a new storage container must be created.

**FIGURE 7:** Virtual datastore to storage container.

For in-depth information about vSphere Virtual Volumes and its components, please visit [vmware.com/products/vsphere.virtual-volumes](http://vmware.com/products/vsphere.virtual-volumes).
Protocol endpoints

Although storage systems manage all aspects of Virtual Volumes, vSphere hosts have no direct access to Virtual Volumes on the storage side. Instead, vSphere hosts use a logical I/O proxy, called the protocol endpoint, to communicate with Virtual Volumes and virtual disk files that Virtual Volumes encapsulate.

vSphere hosts use protocol endpoints to establish a data path on demand from VMs to their respective Virtual Volumes.

Protocol endpoints are compatible with all SAN/NAS industry-standard protocols:

- iSCSI
- NFS v3
- Fibre Channel
- Fibre Channel over Ethernet

Protocol endpoints are set up and configured by storage and vSphere administrators.

FIGURE 8: Protocol endpoints.
vSphere Virtual Volumes requirements

Software
The use of vSphere Virtual Volumes requires the following software components:

• vCenter Server Appliance™ 6.0 or vCenter Server 6.0 for Windows
• VMware ESXi™ 6.0
• vSphere Web Client

Hardware
The use of vSphere Virtual Volumes requires the following hardware components:

• Any server that certified for vSphere 6.0 listed on the VMware compatibility guide
• A third-party storage array system that supports vSphere Virtual Volumes and is able to integrate with vSphere through VASA
• Depending on the specific vendor implementation, a storage array system may or may not require a firmware upgrade to support vSphere Virtual Volumes; check with your storage vendor for detailed information and configuration procedures

License
The use of vSphere Virtual Volumes requires one of the following licenses:

• Standard
• Enterprise Plus
Configuring vSphere Virtual Volumes

The configuration of vSphere Virtual Volumes requires that both the storage system and the vSphere environment are prepared correctly. From a storage perspective, the Virtual Volumes required components such as the protocol endpoints, storage containers, and storage profiles must be configured.

The procedure for configuring the Virtual Volumes components on the storage system varies based on the storage vendor implementation and can be potentially different based on the array brand and model.

For detailed information on the procedures to configure the Virtual Volumes required components, refer to the storage system’s documentation or contact your storage vendor.

Configuration guidelines

The following requirements must be satisfied before enabling Virtual Volumes.

Storage

• The storage system must be Virtual Volumes compatible and able to integrate with vSphere 6.0 through VASA 2.0.
• A storage vendor provider must be available. If the vendor provider is not available as part of the storage system, a vendor provider appliance must be deployed.
• The protocol endpoints, storage containers, and storage profiles must be configured on the storage system.

vSphere

• Make sure to follow the appropriate vendor and VMware guidelines to set up the appropriate storage solution that will be used (Fibre Channel, Fibre Channel over Ethernet, iSCSI, or NFS). This may require the installation and configuration of physical or software storage adapters on vSphere hosts.
• Synchronize the time of all storage components with vCenter Server and all vSphere hosts. It is recommended to utilize Network Time Protocol (NTP) for the synchronization.

Configuration procedures

The following procedures must be performed to configure Virtual Volumes in vSphere. The procedures focus on tasks specific to vSphere and workflow of the components:

• vSphere storage time synchronization
• Storage provider (VASA) registration
• Virtual datastore creation for Virtual Volumes

vSphere storage time synchronization

Before enabling Virtual Volumes, it is recommended for all the vSphere hosts and vCenter Server instances to have their time synchronized. VMware recommends the use of a network time server for all the systems to maintain accurate time keeping.

vCenter Server time synchronization

Perform the required procedure to configure the vCenter Server instances to utilize a time synchronization service suitable to the version of vCenter Server being used (Windows or Linux).

ESXi host time synchronization

Perform the following steps on vSphere hosts that will be utilized for Virtual Volumes. This procedure may be automated with PowerCLI and other command line utilities.
vSphere host time synchronization configuration procedure
1. Select the host in the vSphere inventory.
2. Click the Configure tab.
3. Select Time Configuration in the System section.
4. Click Edit and set up the NTP server:
   a. Select Use Network Time Protocol (Enable NTP client).
   b. Set the NTP Service Startup Policy to Start and stop with host.
   c. Enter the IP or URL addresses of the NTP server(s) to synchronize.
5. Click OK. The host will now synchronize with the NTP server(s).
Storage provider registration
To create a virtual datastore for Virtual Volumes, a storage container must exist in vSphere, and a communication link must be established between the vCenter Server instance and the storage system. The VASA provider exports the storage system’s capabilities and presents them to the vCenter Server instances as well as the vSphere hosts via VASA.

Storage provider registration procedure
In the event the storage provider is not implemented as a hardware component of the storage system, verify that a VASA provider appliance has been deployed and obtain its credentials from the storage administrator.

1. Browse to vCenter Server in the vSphere Client.
2. Click the Configure tab and click Storage Providers.
3. Click +Add to add a new storage provider.

4. Type the connection information for the storage provider, including the name, URL, and credentials.

5. (Optional) Select the Use storage provider certificate option and specify the certificate’s location to direct vCenter Server to the storage provider certificate. If you do not select this option, a thumbprint of the certificate is displayed. You can check the thumbprint and approve it.
6. Click OK to complete the registration.

![Storage Providers](image)

### FIGURE 13: Storage provider successfully registered.

At this point, the vCenter Server instance has been registered with the VASA provider and established a secure SSL connection.

**Note:** Storage providers can also be automatically configured through the storage system’s vSphere Client plug-in or from the storage system’s UI when registering a vCenter.

### Virtual datastore creation procedure

To create a virtual datastore for Virtual Volumes, use the New Datastore wizard from the vSphere Client.

1. Select a host in the vSphere inventory.
2. Right-click and browse to the storage menu.
3. Click the New Datastore option.
4. Type a datastore name:
   a. Ensure the name utilized is not a duplicate of another datastore name in the vCenter Server inventory.
   b. If the same datastore will be mounted on multiple hosts, the name of the datastore must be the same across all the hosts.

![Add New Datastore wizard](image)

### FIGURE 14: Add New Datastore wizard.
5. Select VVol as the virtual datastore type.

**FIGURE 15:** Datastore type.

6. Select a backing storage container from the list of storage containers.

**FIGURE 16:** Storage container selection.
7. Click Next to review the configuration options and click Finish.

**FIGURE 17:** Virtual datastore mapping to a storage container.

After creating the virtual datastore, other datastore operations such as renaming, browsing, mounting, and unmounting the datastore may be performed. To mount the virtual datastores to other hosts, use the Mount Datastore to Additional Hosts wizard in the vSphere Client.
Mounting the virtual datastore onto multiple hosts

Once a virtual datastore is created and mounted onto a single host, the virtual datastore configuration procedure will not work. This is because a virtual datastore is already mapped to a storage container. To mount the virtual datastore, use the mounting virtual datastore to multiple hosts procedure.

This procedure is performed from the Storage view of the vSphere Client.

1. Navigate to the Storage view tab from the vSphere Client.
2. Right-click on the desired virtual datastore and select Mount Datastore to Additional Hosts.

![FIGURE 18: Mount datastore to additional hosts.](image-url)
3. Select the available hosts to mount the virtual datastore and click OK.

![Figure 19: Multiple hosts selection.](image)

The virtual datastore should be automatically mounted to all the selected hosts. The Connectivity with Hosts view—located in the virtual datastore settings under the Configure tab of the Storage view in the vSphere Client—can validate the action was successful.

![Figure 20: Virtual datastore multiple host mount validation.](image)

**Mapping storage capabilities to VM storage policies**

The storage capabilities are configured and managed on the storage systems by the storage admin. Storage capabilities are presented to vSphere via VASA in the form of data services and unique storage system features.

A vSphere admin maps the storage capabilities presented to vSphere and organizes them into a set of rules designed to capture the quality of service (QoS) requirements for VMs and its application. These rules are saved to vSphere in the form of a VM storage policy.

Virtual Volumes utilizes a VM storage policy to manage related operations such as placement decision, admission control, QoS compliance monitoring, and dynamic storage resource allocation.

**Mapping storage capabilities to VM storage policies procedure**

Once all the Virtual Volumes related components have been configured in the infrastructure, a vSphere admin needs to define storage requirements and storage service for a VM and/or virtual disks.
To satisfy the VM service requirements, a VM storage policy needs to be created in vSphere.

Before proceeding with the definition and creation of a VM storage policy, verify that the vendor provider is available and online.

1. Click Menu and select Policies and Profiles from the vSphere Client home screen.

   ![Policy and profiles](image)

   **FIGURE 21:** Policy and profiles.

2. Click the Create VM Storage Policy icon.

   ![Create VM storage policy](image)

   **FIGURE 22:** Create VM storage policy.

3. Select the vCenter Server instance if more than one vCenter is available.
4. Enter a name and description for the storage policy.
5. Select the vendor provider on the Rule-Set 1 window for the storage system that is registered with vSphere from the Rule based and data services dropbox:
   a. The page expands to show the capabilities reported by the storage system.
   b. Add the necessary capabilities and specify a value, if needed.
   c. Make sure the value provided is within the range of values advertised by the capability profile of the storage system.
   d. (Optional) Add tag-based capabilities.
6. Review the list of datastores that match the VM storage policy and click Next.

7. Verify the VM storage policy configuration settings and click Finish.

To be eligible, a virtual datastore needs to satisfy all rule sets defined within the VM storage policy.

Make sure that the storage system’s storage containers meet the requirements set in the VM storage policy and appear on the list of compatible datastores. The VM storage policy should have now been added to the list and can be applied to VMs and its virtual disks.
Virtual machine creation

Once the vSphere infrastructure and storage systems are ready and their respective policies and capabilities have been configured and defined, a vSphere admin can start deploying VMs onto virtual datastores for Virtual Volumes.

To create a new VM and deploy it onto a virtual datastore for Virtual Volumes, use the following procedure.

1. Select any VM parent object in the vSphere inventory:
   a. Data center
   b. Cluster
   c. Host
   d. Resource pool
   e. Folder

FIGURE 26: Create a new VM.

2. Right-click any of these objects and choose New Virtual Machine.

3. Select the Create a new virtual machine option in the New Virtual Machine wizard and click Next.

FIGURE 27: Virtual machine creation type.
4. Enter a name for the VM, select a location for it, and click Next:
   a. Data center
   b. VM folder

5. Select the compute resource for the VM and click Next:
   a. Cluster
   b. Host
   c. vApp
   d. Resource pool

![FIGURE 28: Select the compute resource.](image)

6. Choose a VM storage policy to configure the VM storage requirements for a virtual datastore. Then, select the compatible datastore that meets the storage requirements of the chosen policy and click Next.

![FIGURE 29: Select storage with a VM storage policy.](image)
Note: The select VM storage policy option is available on all VM provisioning-related operations such as deploy from template, clone an existing virtual machine, clone a virtual machine to template, and clone template to template.

7. Select the host compatibility level for the VM and click Next.
8. Choose the guest OS family and guest OS version that will be installed on the VM and click Next.
9. Customize the VM hardware as needed, then click Next.
10. Review the VM configuration, including the VM storage policy selected, for accuracy, then click Finish.

Virtual machine migration with vSphere Storage vMotion
With vSphere Storage vMotion®, VMs and their disk files may be migrated from a VMFS or NFS type virtual datastore to a Virtual Volumes type of virtual datastore. The migration operation can be performed between virtual datastores located on the same storage system or on different storage systems.

The migration operations may be performed while the VMs are powered on or off.

As part of the migration operation, you select the virtual disk format and the VM storage policy, and choose to place the VM and all its disks in a single location or select separate locations for the VM configuration file and each virtual disk. The VM does not change execution hosts during a migration with Storage vMotion.
To migrate a new VM onto a Virtual Volumes datastore, use the following procedure.

1. Select any VM in the vSphere inventory and select Migrate.

   ![Virtual machine migration](image)
   
   **FIGURE 31:** Virtual machine migration.

2. Select a migration type:
   a. Change compute resource only
   b. Change storage only
   c. Change both compute and storage

   In this case, select Change storage only and click Next.

   ![Storage vMotion migration options](image)
   
   **FIGURE 32:** Storage vMotion migration options.
3. Select the destination storage, virtual disk format, VM storage policy, and suitable virtual datastore, and click Next.

![Figure 33: Storage vMotion storage target.](image)

4. Review the migration details—including the virtual datastore target, VM storage policy, and disk format settings selected—for accuracy, then click Finish.

![Figure 34: Storage vMotion configuration overview.](image)
vSphere Virtual Volumes interoperability

vSphere Enterprise features
The following vSphere 6.x Enterprise Edition™ features are supported by Virtual Volumes:

• Storage Policy-Based Management
• Thin Provisioning
• Linked Clones
• Native Snapshots
• NFS v3.x
• View Storage Accelerator (CBRC)
• vMotion
• Storage vMotion
• vSphere SDK (VC API)
• vSphere Web Client
• Host Profiles/Stateless
• vSphere High Availability
• XvMotion
• vSphere Auto Deploy™

VMware products and solutions
The following VMware products and solutions provide support and interoperability for Virtual Volumes:

• VMware vSphere 6.x
• VMware vRealize® Automation™ 6.2
• VMware Horizon® 6.1
• VMware vSphere Replication™ 6.x
• VMware vSAN™ 6.x

Note: Virtual Volumes storage containers can be presented to hosts that are members of a VMware vSAN cluster. VMs can have VMDKs simultaneously stored on both Virtual Volumes storage containers and a vSAN datastore.
vSphere Virtual Volumes CLI commands
The ESXCLI command line framework has been updated to include a Virtual Volume module. All of the new Virtual Volumes ESXCLI commands are grouped under the storage Virtual Volume namespace.

Virtual Volumes ESXCLI namespaces
The Virtual Volumes namespace contains multiple sets of commands. Each namespace focuses on a different operating function of the Virtual Volumes related components. There are five available namespaces:

- **storagecontainer** – Operations to create, manage, and remove Virtual Volume storage containers
- **daemon** – Operations pertaining to a Virtual Volumes daemon
- **protocolendpoint** – Operations on Virtual Volumes protocol endpoints
- **vasacontext** – Operations on the Virtual Volumes and VASA context
- **vasaprovider** – Manages Virtual Volumes VASA provider operations

ESXCLI storage Virtual Volume command line syntax samples
```
esxcli storage vvol -h
```
Usage: esxcli storage vvol {cmd} [cmd options]
Available Namespaces:

- **storagecontainer** Operations to create, manage, remove and restore VVol StorageContainers.
- **daemon** Operations pertaining to VVol daemon.
- **protocolendpoint** Operations on VVol Protocol EndPoints.
- **vasacontext** Operations on the VVol VASA context.
- **vasaprovider** Manage VVol VASA Provider Operations.

Virtual Volumes ESXCLI namespace commands
The Virtual Volumes storagecontainer namespace commands provide the ability to list the storage containers mapped to an ESXi host, as well as the ability to scan for abandoned Virtual Volumes within storage containers.

ESXCLI storage Virtual Volume storagecontainer command line syntax samples
```
esxcli storage vvol storagecontainer -h
```
Usage: esxcli storage vvol storagecontainer {cmd} [cmd options]
Available Namespaces:

- **abandonedvvol** Operations on Abandoned Virtual Volumes.

Available Commands:

- **list** List the VVol StorageContainers currently known to the ESX host.

The abandonedvvol namespace identifies Virtual Volumes that have been abandoned when a failure to delete event happens, such as a failure to delete a swap Virtual Volume
during a VM power-off operation through a particular path. This behavior typically happens when there are communication issues with the vendor/VASA provider.

In this scenario, instead of failing the VM power-off operation, the system makes note of that Virtual Volume on a per-VM-namespace basis onto an abandon Virtual Volumes tracking file so it can be deleted when the vendor/VASA provider is back online. A periodic thread tries to delete such abandoned Virtual Volumes.

The scan option allows the initiation of a background scan of a respective Virtual Volumes datastore, searching for abandoned Virtual Volumes. The operation goes over all the config Virtual Volumes, looking for the abandoned Virtual Volumes tracking files and tries to delete them.

The successful initiation of the scan doesn't indicate that the operation succeeded or failed. This operation might take long time to complete, as we don't scan all the config Virtual Volumes at once to avoid putting load on the vendor/VASA provider for a non-important operation such as garbage collecting old Virtual Volumes.

**ESXCLI storage Virtual Volume storagecontainer abandonedvvol syntax samples**

```bash
esxcli storage vvol storagecontainer abandonedvvol scan -p
eqlDatastore
true
```

List provides the ability to display or list the number of virtual datastores and details for Virtual Volumes that are known to a particular vSphere host.

**ESXCLI storage Virtual Volume storage container list syntax sample**

```bash
esxcli storage vvol storage container list
eqlDatastore
```

```
<table>
<thead>
<tr>
<th>StorageContainer Name: eqlDatastore</th>
<th>UUID: vvol:6090a0681067ae78-2e48c5020000a0f6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Array: com.dell.storageprofile.equallogic.std:eqlgrp1</td>
<td>Size(MB): 1048590</td>
</tr>
<tr>
<td>Free(MB): 972540</td>
<td>Accessible: true</td>
</tr>
<tr>
<td>Default Policy:</td>
<td>engDatastore</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>StorageContainer Name: engDatastore</th>
<th>UUID: vvol:6090a06810770d5b-cc4ad5d7a1042074</th>
</tr>
</thead>
<tbody>
<tr>
<td>Array: com.dell.storageprofile.equallogic.std:eqlgrp1</td>
<td>Size(MB): 4194315</td>
</tr>
<tr>
<td>Free(MB): 4173930</td>
<td>Accessible: true</td>
</tr>
<tr>
<td>Default Policy:</td>
<td>dbDatastore</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>StorageContainer Name: dbDatastore</th>
<th>UUID: vvol:6090a0681077bdce-8b4b1515a2049013</th>
</tr>
</thead>
<tbody>
<tr>
<td>Array: com.dell.storageprofile.equallogic.std:eqlgrp1</td>
<td>Size(MB): 1024005</td>
</tr>
<tr>
<td>Free(MB): 1009635</td>
<td>Accessible: true</td>
</tr>
<tr>
<td>Default Policy:</td>
<td></td>
</tr>
</tbody>
</table>
The daemon in the namespace is utilized to unbind Virtual Volume operations from all vendor/VASA providers that are known to a particular vSphere host.

The unbindall option is utilized to unbind all Virtual Volumes from all the vendor/VASA providers known to a particular ESXi host. This operation is performed for testing purposes or to force the cleanup of all Virtual Volumes data paths.

ESXCLI storage Virtual Volume storagecontainer daemon unbindall syntax sample
Usage: esxcli storage vvol daemon unbindall [cmd options]

Description:
unbindall
Unbind all virtual Volumes from all VPs known to the ESX host.

Cmd options:
esxcli storage vvol daemon unbindall

The protocolendpoint namespace commands provide the ability to list all the information with regards to the protocol endpoints configuration to a vSphere host.

List provides the ability to display or list the number of protocol endpoints and their configuration details to a particular ESX host.

esxcli storage vvol protocolendpoint
Usage: esxcli storage vvol protocolendpoint {cmd} [cmd options]

Available Commands:
list List the VVol Protocol EndPoints currently known to the ESX host.

esxcli storage vvol protocolendpoint list

naa.6090a0681077ad11863e05020000a061
Host Id: naa.6090a0681077ad11863e05020000a061
Array Id: com.dell.storageprofile.equallogic.std:eqlgrp1
Type: SCSI
Accessible: true
Configured: true
Lun Id: naa.6090a0681077ad11863e05020000a061
Remote Host:
Remote Share:
Storage Containers: 6090a068-1067-ae78-2e48-c5020000a0f6

The vasacontext namespace command provides the ability to get the vCenter Server UUID for which the vendor/VASA provider is currently registered.

The get option is utilized to get the Virtual Volume VASA context or vCenter Server UUIDs.

esxcli storage vvol vasacontext -h
Usage: esxcli storage vvol vasacontext {cmd} [cmd options]
Available Commands:

get  Get the VVol VASA Context (VC UUID).

esxcli storage vvol vasacontext get 5742ead8-0695-48bd-9ae4-7416164423ef

The vasaprovider namespace command provides the ability to list the vendor/VASA providers currently registered onto a particular ESXi host.

The list option is utilized to list all the vendor/VASA providers and their information details that are registered to a particular ESXi host.

esxcli storage vvol vasaprovider -h

Usage: esxcli storage vvol vasaprovider {cmd} [cmd options]

Available Commands:

list  List the VASA Providers registered on the host.

esxcli storage vvol vasaprovider list

Dell Equallogic VASA Provider
   VP Name: Dell Equallogic VASA Provider
   URL: https://10.144.106.39:8443/vasa-version.xml
   Status: online
   Arrays:

Array Id: com.dell.storageprofile.equallogic.std:eqlgrp1
Is Active: true
Priority: 0
Reference

vSphere Virtual Volumes resources
vSphere Virtual Volumes Product Page

Product documentation
vSphere Virtual Volumes solution overview
vSphere Virtual Volumes FAQ
Virtual Volumes: Getting Started Guide

Documentation
Virtual Volumes docs
Virtual Volumes references

Presentations
VMworld

About the authors

Rawlinson Rivera was a principal architect in the Storage and Availability Business Unit at VMware. He specialized in cloud enterprise architectures, software-defined storage, hyperconverged infrastructures, and business continuity/disaster recovery solutions, with a focus on storage solutions such as vSAN and vSphere Virtual Volumes.

Follow Rawlinson on Twitter: @PunchingClouds

Jason Massae is a technical marketing architect in the Storage and Availability Business Unit at VMware, focusing on core/external storage, vSphere Virtual Volumes, and occasionally vSAN. Jason has more than 20 years in the IT industry concentrating on data center architecture and operations, and specializing in virtualization for large enterprises. Jason now focuses on delivering relevant, consumable content to the masses from real-world experiences.

Follow Jason on Twitter: @jbmassae