Introduction

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.

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 are focused around operational efficiencies and flexible consumption models.

- Virtual Volumes simplifies storage operations by automating manual tasks and eliminating operational dependencies between the vSphere Admin and the Storage Admin. Provisioning is faster, and change management is simpler as the new operational model is built upon policy-driven automation.

- 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 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 virtual machines 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 vSphere administrator.

Since a single, homogeneous storage pool would potentially contain many different 
applications and virtual machines; this approach resulted in needless complexity 
and inefficiency. vSphere administrators could not easily specify specific 
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 in the future, usually resulting in 
the overprovisioning of resources.

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

![Figure 2: vSphere Virtual Volumes Operational Model](image)

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 virtual machines, which might have different requirements. With 
the traditional approach, differentiation on a per virtual machine level is difficult.

The Virtual Volumes functionalities allows for the differentiation of virtual machine 
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 virtual machines, making storage virtual machine centric.

Virtual Volumes map virtual disks and their respective components directly to objects, called virtual volumes, 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 that are 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 is a summarized description and definition of the key components of vSphere Virtual Volumes:

**Virtual Volumes (VVols)**

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

There are five different types of Virtual Volumes object types and each of them map to a different and specific virtual machine file.

![Figure 3: vSphere Virtual Volumes Object Types](image)

- **Config** – VM Home, Configuration files, logs
- **Data** – Equivalent to a VMDK
- **Memory** – Snapshots
- **SWAP** – Virtual machine memory swap
- **Other** – vSphere solution specific object
Vendor Provider (VP)

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

ESXi hosts and vCenter Server connect to the Vendor 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 SPBM.

Vendor Providers are typically setup and configured by the vSphere administrator in one of two ways:

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

![Vendor Provider Diagram](image-url)
Storage Container (SC)

Unlike traditional LUN and NFS based vSphere storage, the Virtual Volumes functionality does not require preconfigured volumes on a 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 setup and configured by storage administrators.

Containers are used to define:

- Storage capacity allocations and restrictions
- Storage policy settings based on data service capabilities on a per virtual machine basis
**Virtual Datastore**

A Virtual Datastore represents a storage container in a vCenter Server instance and the vSphere Web 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 Volumes created VMDKs are created.

Virtual Datastores are typically setup and configured by vSphere administrators.

![Figure 6: Virtual Datastore](image)

**Note:** For in-depth information about vSphere Virtual Volumes and its components please refer to the official vSphere Virtual Volumes product page [http://www.vmware.com/products/virtual-volumes](http://www.vmware.com/products/virtual-volumes).
Protocol Endpoints (PE)

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

ESXi uses protocol endpoints to establish a data path on demand from virtual machines to their respective virtual volumes.

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

- iSCSI
- NFS v3
- Fiber Channel (FC)
- Fiber Channel over Ethernet (FCoE)

Protocol Endpoints are setup and configured by Storage administrators.

Figure 7: Protocol Endpoint
vSphere Virtual Volumes Requirements

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

- vCenter Server 6.0 Appliance (VCSA) or vCenter Server 6.0 for Windows
- ESXi 6.0
- vSphere Web Client

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

- Any Server that is certified for vSphere 6.0 that is listed on the VMware compatibility guide.
- A third party storage array system that supports vSphere Virtual Volumes and able to integrate with vSphere through the VMware APIs for Storage Awareness.
- Depending on the vendor specific implementation, storage array system may or may not require a firmware upgrade in order to support vSphere Virtual Volumes. Check with your storage vendor for detailed information and configuration procedures.

License
The use of vSphere Virtual Volumes requires the following license:

- Standard
- Enterprise Plus
Configuring vSphere Virtual Volumes

The configuration of 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 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 the VMware APIs for Storage Awareness (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

- Follow the appropriate guidelines to setup the appropriate storage solution that will be used, Fiber Channel, FCoE, iSCSI, or NFS. This may required the installation and configuration of physical or software storage adapters on ESXi hosts.
- Synchronize the time of all storage components with vCenter Server and all ESXi hosts. It is recommended to utilize Network Time Protocol (NTP) for the synchronization.
Configuration Procedures

In order to configure Virtual Volumes in vSphere 6.0 the procedures defined below must be performed. The procedures are focused on vSphere specific tasks and workflow of the components:

- vSphere Storage Time synchronization
- Storage Provider registration
- Creating a Virtual Datastore for Virtual Volumes

vSphere Storage Time Synchronization

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

vCenter Server Time Synchronization

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

ESXi host Time Synchronization

Perform the following steps on ESXi host that will utilize for Virtual Volumes. This procedure could be automated with PowerCLI and other command line utilities.

ESXi Host Time Synchronization Configuration Procedure

1. Select the host in the vSphere inventory.
2. Click the Manage tab and click Settings.
3. In the System section, select Time Configuration.
4. Click Edit and set up the NTP server.
Figure 8: ESXi Host Time Configuration Settings

a. Select Use Network Time Protocol (Enable NTP client).
b. Set the NTP Service Startup Policy.
c. Enter the IP addresses of the NTP servers to synchronize with.
d. Click Start or Restart in the NTP Service Status section.

Figure 9: NTP Service Configuration

5. Click OK. The host synchronizes with the NTP server.
6. Click the Manage tab and click Settings.
7. In the System section, select Time Configuration.
8. Click Edit and set up the NTP server.
   a. Select Use Network Time Protocol (Enable NTP client).
   b. Set the NTP Service Startup Policy.
   c. Enter the IP addresses of the NTP servers to synchronize with.
   d. Click Start or Restart in the NTP Service Status section.
9. Click OK. The host synchronizes with the NTP server.
Storage Provider Registration

In order to create a Virtual Datastore for Virtual Volumes a storage container must exist in vSphere 6.0 a communication link must be established between the vCenter Server instance, and the storage system. The VASA provider export the storage systems capabilities and present them to the vCenter Server instances as well as the ESXi hosts via the VASA APIs.

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 Web Client navigator.
2. Click the Manage tab, and click Storage Providers.
3. Click the Register a new storage provider icon.
4. Type the connection information for the storage provider, including the name, URL, and credentials.

Figure 10: Storage Provider Manual Registration

5. (Optional) To direct vCenter Server to the storage provider certificate, select the Use storage provider certificate option and specify the certificate’s location. 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.
At this point the vCenter Server instance has been registered with the VASA provider and established a secure SSL connection with it.

**Note:** Storage Providers can also be automatically configured through the storage systems vSphere Web Client plug-ins.

**Virtual Datastore Creation Procedure**

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

1. Select the 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 instance inventory.
   b. If the same datastore will be mounted on multiple hosts, the name of the datastore must be the same across all of the hosts.
5. Select the VVol as the virtual datastore type.
6. From the list of storage containers, select a backing storage container.

7. Click next to review the configuration options and click Finish.
After the virtual datastore is created, other datastore operations such as renaming, browsing, mounting and unmounting the datastore can be performed. To mount the virtual datastores onto other hosts use the Mount Datastore to Additional Hosts wizard in the vSphere Web 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 due to the fact that a virtual datastore is already mapped to a storage container. In order to mount the virtual datastore onto multiple hosts:

1. From the vSphere Web Client navigate to the Storage view tab and select it.
2. Right click on the desired virtual datastore and select Mount Datastore to Additional Hosts.
3. Select the available hosts to mount the Virtual Datastore and click OK.

---

**Figure 15: Mount Datastore to Additional Hosts**

**Figure 16: Multiple Hosts Selection**
The virtual datastore should be automatically mounted onto all the selected hosts. Looking at the Connectivity with Hosts can validate the action was successful. The Connectivity with Host view is located in the virtual datastore settings under the manage tab of the storage view in the vSphere Web Client.

![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 admins. Storage capabilities are presented to vSphere via the VASA APIs 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 that are designed to capture the quality of service requirements for virtual machines and their application. These rules are saved onto vSphere in the form of a VM Storage policy.

Virtual Volumes functionalities utilize VM Storage Policy for management-related operations such as placement decision, admission control, quality of service 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 throughout the infrastructure, a vSphere admin needs to define storage requirements and storage service for a virtual machine and its virtual disks. In order to satisfy the virtual machines service requirements a VM Storage Policies need to be created in vSphere.

Before proceeding with definition and creating of a VM Storage Policies verify that the vendor provider is available and online.

1. From the vSphere Web Client Home screen, click on VM Storage Policies.
2. Click the Create a new Storage Policy icon.
3. Select the vCenter Server instance.
4. Type a name and description for the storage policy.
5. On the Rule-Set 1 window, select the vendor provider for the storage system that is registered with vSphere from the Rule based and data services drop-box.
   a. The page expands to show the capabilities reported by the storage system
   b. Add the necessary capabilities and specify its value
   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.

![Figure 4: List of virtual Datastores Compliant with VM Storage Policy](image)

7. Verify the VM Storage Policy configuration settings and click Finish.

![Figure 5: Policy Setting Confirmation](image)

To be eligible, a virtual datastore does not need to satisfy all rule sets defined within the VM Storage Policy. A Virtual Datastore must satisfy at least one rule set and all rules within the set.

Make sure that the storage systems storage containers meets the requirements set in the VM Storage Policy and appears on the list of compatible datastores. The VM Storage Policy should have now been added to the list and can be applied to virtual machines and its virtual disks.
Virtual Machine Creation

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

To create a new virtual machine and deploy it onto a virtual datastore for Virtual Volumes follow the procedure listed below.

1. Select any virtual machine parent object in the vSphere inventory.
   a. Datacenter
   b. Cluster
   c. Host
   d. Resource pool
   e. Folder

Figure 6: Create New Virtual Machine

2. Right click any of the objects listed above and choose new Virtual Machine.

3. Once the New Virtual Machine wizard opens select the Create a new Virtual Machine option and click Next.
4. Enter a name for the virtual machine and select a location for the Virtual Machine and click Next.
   a. Datacenter
   b. VM folder
5. Select the compute resource for the New Virtual Machine deployment operation and click Next.
   a. Cluster
   b. Host
   c. vApp
   d. Resource Pool
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.

**Note:** The select VM Storage policy option is available on all virtual machine provisioning related operations such as deploy from template, clone an existing virtual machine, clone a virtual machine to template, clone template to template.

7. Select the host compatibility level for the Virtual Machine and click next
8. Choose the Guest OS Family and Guest OS version that will be installed on the Virtual Machine and click Next.
9. Customized the virtual machine hardware as needed then click Next.
10. Review the virtual machine configuration including the VM Storage Policy selected for accuracy the click Finish.

![Virtual Machine Configuration Overview](image)

**Virtual Machine Migration with Storage vMotion**

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

The migration operations can be performed while the virtual machines are powered on or powered off.

As part of the migration operation, you select the virtual disk format, VM Storage Policy and choose to place the virtual machine and all its disks in a single location, or select separate locations for the virtual machine configuration file and each virtual disk. The virtual machine does not change execution host during a migration with Storage vMotion.

To migrate a new virtual machine onto a Virtual Volumes datastore follow the procedure listed below.
1. Select any virtual machine in the vSphere inventory and select Migrate.

![Figure 11: Virtual Machine Migration](image1)

2. Select one of the migration types options, in this case Change storage only and click Next.
   a. Change compute resource only
   b. Change storage only
   c. Change both compute and storage

![Figure 12: SvMotion Migration Options](image2)
3. Select the destination storage, virtual disk format, VM Storage Policy, and suitable virtual datastore and click Next.

![Figure 13: SvMotion Storage Target](image1)

4. Review the migration details including the virtual datastore target, VM Storage Policy, and disk format settings selected for accuracy the click Finish.

![Figure 14: SvMotion Configuration Overview](image2)
vSphere Virtual Volumes Interoperability

vSphere Enterprise Features

The following table highlights the vSphere enterprise features that are available in the vSphere 6.0 are supported with Virtual Volumes.

<table>
<thead>
<tr>
<th>vSphere 6.0 Supported Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage Policy Based Management (SPBM)</td>
</tr>
<tr>
<td>Thin Provisioning</td>
</tr>
<tr>
<td>Linked Clones</td>
</tr>
<tr>
<td>Native Snapshots</td>
</tr>
<tr>
<td>NFSv3.x</td>
</tr>
<tr>
<td>View Storage Accelerator (CBRC)</td>
</tr>
<tr>
<td>vMotion</td>
</tr>
<tr>
<td>Storage vMotion</td>
</tr>
<tr>
<td>vSphere SDK (VC API)</td>
</tr>
<tr>
<td>vSphere Web Client</td>
</tr>
<tr>
<td>Host Profiles / Stateless</td>
</tr>
<tr>
<td>vSphere HA</td>
</tr>
<tr>
<td>XvMotion</td>
</tr>
<tr>
<td>vSphere Auto Deploy</td>
</tr>
</tbody>
</table>

Table 1: vSphere 6.0 Enterprise Features Supported by Virtual Volumes
VMware Products and Solutions

The following table highlights the products and solutions that are available, which provide support and interoperability for are Virtual Volumes.

<table>
<thead>
<tr>
<th>VMware Supported Products and Solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>VMware vSphere 6.0</td>
</tr>
<tr>
<td>VMware vRealize Automation 6.2</td>
</tr>
<tr>
<td>VMware Horizon 6.1</td>
</tr>
<tr>
<td>VMware vSphere Replication 6.0</td>
</tr>
<tr>
<td>VMware Virtual SAN 6.0</td>
</tr>
</tbody>
</table>

Table 2: VMware Products and Solutions Supported by Virtual Volumes

**Note:** Virtual Volumes storage containers can be presented to hosts that are members of a VMware Virtual SAN cluster. Virtual machines can have VMDKs simultaneously stored on both Virtual Volumes Storage Containers and a Virtual SAN 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, VVol namespace.

VVols ESXCLI Namespaces

The VVols namespace contains multiple sets of commands. Each namespace is focus on a different operating function of the Virtual Volumes related components. There are five available namespaces and their respective description are listed below:

- **storagecontainer** – Operations to create, manage, remove VVol Storage Containers
- **deamon** – Operations pertaining to VVol deamon
- **protocolendpoint** – Operations on VVol Protocol Endpoints.
- **vasacontext** – Operations on the VVol and VASA context.
- **vasaprovider** – Manage VVol VASA Provider Operations

esxcli storage vvol 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.
```
VVols ESXCLI Namespace Commands

The VVol storagecontainer namespace commands provide the ability to list the storage containers are mapped to an ESXi host. As well as the ability to scan for abandoned VVols within storage containers.

esxcli storage vvol 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.
```

**abandonedvvol** – is a state in which VVols are placed whenever a failure to delete event happens. i.e. failure to delete swap vvol during a VM power-off operation thought 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 VVol on a per-VM-namespace basis onto an abandon VVols tracking file so that it could be deleted later when the Vendor/VASA Provider is back online. A periodic thread tries to delete such abandoned VVols.

**scan option** – this option allows the initiation of a background scan of a respective VVol datastore, searching for abandoned VVols. The operation goes over all the Config-VVols, looking for the abandoned VVols tracking files and tries to delete them.

The successful initiation of the scan doesn’t indicate that the operation succeeded or failed. This is a long running operation that might take long time to complete, as we don't scan all the config-VVols at once to avoid putting load on the Vendor/VASA Provider for a non-important operation like garbage collecting old VVols.
esxcli storage vvol storagecontainer abandonedvvol syntax samples:

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 VVols and that are known to a particular ESX host.

esxcli storage vvol storagecontainer list syntax sample:

esxcli storage vvol storagecontainer list
eqlDatastore
  StorageContainer Name: eqlDatastore
  UUID: vvol:6090a0681067ae78-2e48c5020000a0f6
  Array: com.dell.storageprofile.equallogic.std:eqlgrp1
  Size(MB): 1048590
  Free(MB): 972540
  Accessible: true
  Default Policy:

engDatastore
  StorageContainer Name: engDatastore
  UUID: vvol:6090a06810770d5b-cd4ad5d7a1042074
  Array: com.dell.storageprofile.equallogic.std:eqlgrp1
  Size(MB): 4194315
  Free(MB): 4173930
  Accessible: true
  Default Policy:

dbDatastore
  StorageContainer Name: dbDatastore
  UUID: vvol:6090a0681077bdce-8b4b1515a2049013
  Array: com.dell.storageprofile.equallogic.std:eqlgrp1
  Size(MB): 1024005
  Free(MB): 1009635
  Accessible: true
  Default Policy:

The daemon in the namespace – Is utilized to perform unbind virtual volume operations from all Vendor/VASA Provider that are known to a particular ESX host.
**unbindall** – this option is utilized to unbind all VVols from all the Vendor/VASA Provider known to a particular ESXi Host. This operation is performed for testing purposes or to force the cleanup of all VVols data path.

**esxcli storage vvol 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 the all the information with regards to the Protocol Endpoints configuration to an ESXi 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:
lst List the VVol Protocol EndPoints currently known to the ESX host.

esxcli storage vvol protocolendpoint lst

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.

**get** - this option is utilized to get the VVol VASA context or vCenter Server UUID's.

```
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 that are currently registered onto a particular ESXi host.

**list** - this 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

What’s New: VMware vSphere Virtual Volumes

VMware Blogs

vSphere Virtual Volumes
http://blogs.vmware.com/vsphere/2015/02/vsphere-virtual-volumes.html

vSphere Virtual Volumes Interoperability: VAAI APIs vs VVols

VMworld Presentations

vSphere Virtual Volumes Technical Deep Dive – STO1965
https://www.youtube.com/watch?v=zFmKPJKe3I4
About the Author

Rawlinson Rivera is a Principal Architect working in the Office of the CTO in the Storage and Availability Business Unit at VMware, Inc. He specializes in cloud enterprise architectures, Software-Defined storage, Hyper-converged Infrastructures and business continuity / disaster recovery solutions with focus on storage solutions Virtual SAN, vSphere Virtual Volumes, as well as storage solutions for OpenStack and Cloud-Native Applications. He serves as a partner and trusted adviser to VMware’s customers primarily in the US.

Rawlinson is amongst the few VMware Certified Design Experts (VCDX#86) in the world, and author of multiple books based on VMware and other technologies. He is the owner and main author of virtualization blog punchingclouds.com.

- Follow Rawlinson's blogs:
  
  http://blogs.vmware.com/vsphere/storage
  http://www.punchingclouds.com

- Follow Rawlinson on Twitter:
  
  @PunchingClouds