Dell EMC Unity™ Family

Version 4.3

Configuring VVols

H14975

REV 04

DELL EMC

Published January 2018

Dell EMC
Hopkinton, Massachusetts 01748-9103
1-508-435-1000 In North America 1-866-464-7381
www.DellEMC.com

CONTENTS

Additional resources

As part of an improvement effort, revisions of the software and hardware are periodically released. Therefore, some functions described in this document might not be supported by all versions of the software or hardware currently in use. The product release notes provide the most up-to-date information on product features. Contact your technical support professional if a product does not function properly or does not function as described in this document.

Where to get help
Support, product, and licensing information can be obtained as follows:

Product information
For product and feature documentation or release notes, go to Unity Technical Documentation at: www.emc.com/en-us/documentation/unity-family.htm.

Troubleshooting
For information about products, software updates, licensing, and service, go to Online Support (registration required) at: https://Support.EMC.com. After logging in, locate the appropriate **Support by Product** page.

Technical support
For technical support and service requests, go to Online Support at: https://Support.EMC.com. After logging in, locate **Create a service request**. To open a service request, you must have a valid support agreement. Contact your Sales Representative for details about obtaining a valid support agreement or to answer any questions about your account.

Special notice conventions used in this document

DANGER

Indicates a hazardous situation which, if not avoided, will result in death or serious injury.

WARNING

Indicates a hazardous situation which, if not avoided, could result in death or serious injury.

CAUTION

Indicates a hazardous situation which, if not avoided, could result in minor or moderate injury.

NOTICE

Addresses practices not related to personal injury.

Note

Presents information that is important, but not hazard-related.

CHAPTER 1

Manage VMware virtual volume datastores

This chapter addresses the following topics:

VMware virtual volumes

Virtual Volumes (VVols) are a VMware object type that corresponds to a Virtual Machine (VM) disk, and its snapshots and fast-clones. There are different types of VVol objects, including Config-VVol, Data-VVol (equivalent to VMDK), Memory-VVol, and Swap-VVol.

On the storage system, VVols reside in VVol datastores, also known as storage containers. VVol datastores are another type of VMware datastore, in addition to VMFS and NFS datastores, which allow VVols to map directly to a storage system. Whereas VMware VMFS and NFS datastores are managed and provisioned at the LUN or file system-level, VVol datastores are more granular: VMs or virtual disks can be managed independently. You can create VVol datastores based on one or more underlying storage pools and then allocate a specific portion of the pool to be used for the VVol datastore and its associated VVols.

VMware vSphere 6.0 and later uses Storage Policy-Based Management (SPBM) to define application or VM-specific storage requirements. These storage policies dictate which storage containers are compatible with VVols. A capability profile, configured by the storage administrator, is a set of performance characteristics for a VVol datastore/storage container on the storage system. These characteristics are based on the underlying storage pools and include three categories of capabilities:

- Service level-based provisioning
- Usage tags
- Storage properties
- Host IO limits

Capability profiles are populated through the VMware vStorage API for Storage Awareness (VASA) protocol from the storage system into vSphere or vCenter. These capability profiles map to VMware VVol storage policy profiles. When a storage policy is selected in vSphere or vCenter, only those VVol datastores compatible with these policies will appear as eligible storage containers for the virtual volume.

NAS and SCSI Protocol Endpoints (PEs) are access points for ESXi host I/O communication from VMs to their VVol datastores on the storage system.

VVols workflow

Creating virtual volumes involves several steps in Unisphere. This prepares the storage system for the deployment of virtual volumes from the ESXi host.

Figure 1 Block VVols Workflow

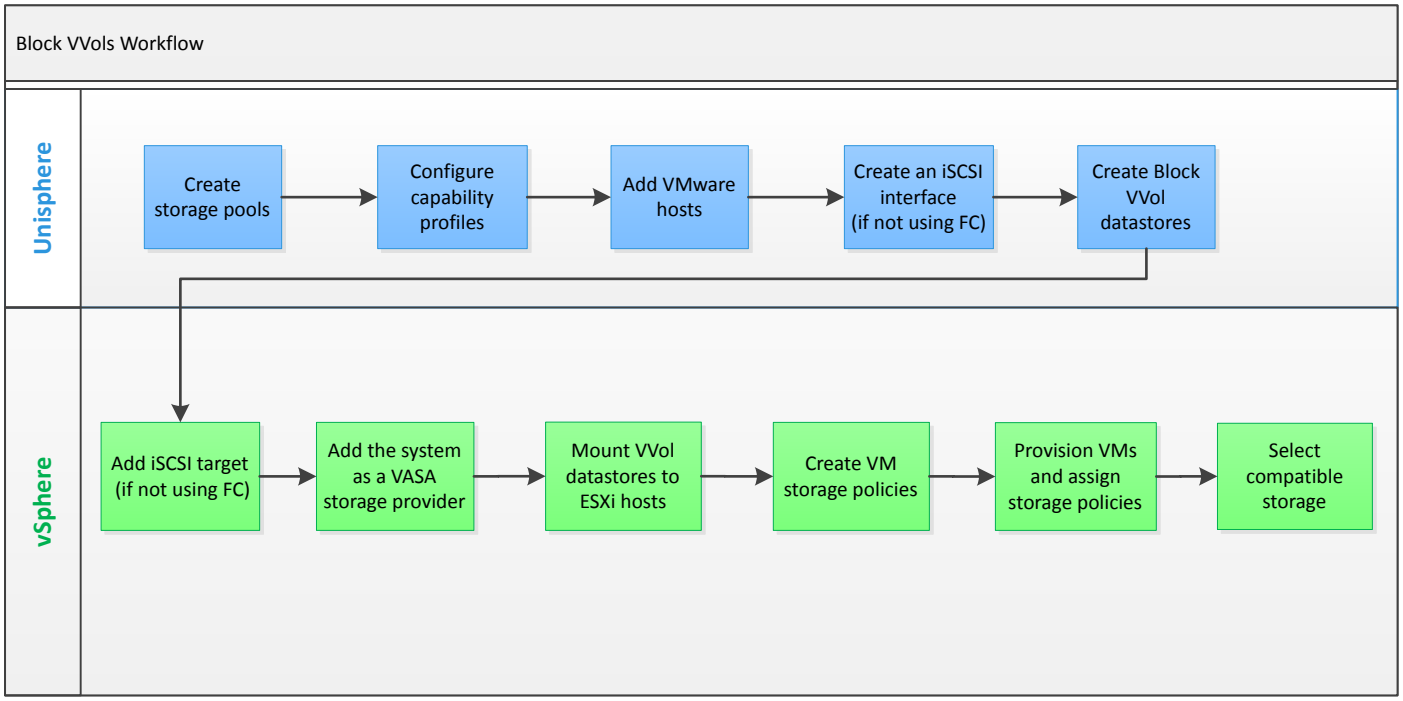
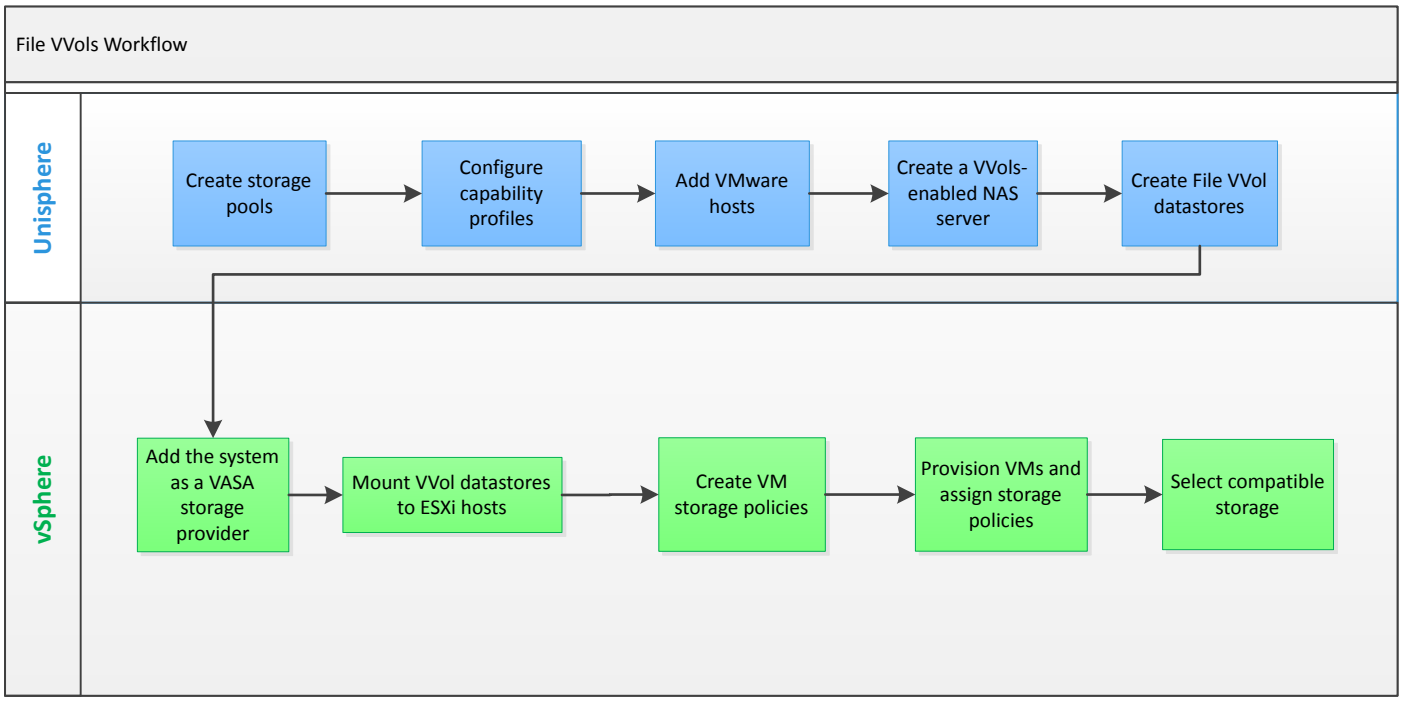


Figure 2 File VVols Workflow



Create a traditional pool in physical deployments

Before you begin

All pools created with All-Flash models running Unity OE version 4.1.x and earlier, and all hybrid and VSA models support traditional pools only. With newer All-Flash models, you can create a traditional pool using the Unisphere CLI and REST API.

Before you create a traditional pool:

- If you have a hybrid model, find out whether the storage system is licensed for FAST Cache. To do this, select the Settings icon, and then select **Storage Configuration** > **FAST Cache**. If the storage system is licensed for FAST Cache, you can choose whether to use it for the pool.
- If you have a hybrid model, find out whether the storage system is licensed for FAST VP. To do this, select the Settings icon, and then select **Storage Configuration** > **FAST VP**. If the storage system is licensed for FAST VP you can choose which storage tiers to add to the pool.
- Decide whether to change the suggested RAID type of the tiers.
- Decide whether to create a capability profile that has capabilities based on the pool configuration. To use the capability profile for VMware VVols, you must assign specific usage tags, which are propagated to the VMware vSphere environment, and can be used in policy profiles. The virtualization administrator and storage administrator should work together to define these tags.

Procedure

1. Under **Storage**, select **Pools**.
2. Select the **Add** icon.
3. Follow the steps in the wizard, taking into account the following considerations:
    - On the **Tiers** screen, you can only select multiple storage tiers if the system is a hybrid model that is licensed to use FAST VP. The wizard displays a maximum usable capacity for each selected tier, which it calculates based on the default RAID configuration. You can optionally change the RAID configuration for all selected tiers.
    - The number and types of drives you can choose is based on the RAID configuration.

Create a pool in virtual deployments

Before you begin

All pools created in virtual deployments are traditional pools.

Before you create a traditional pool:

- From the storage administrator, obtain information about the underlying characteristics of the drives to use in the pool. You will use this information to assign tiers to the virtual disks that do not already have them assigned. You can create a multi-tier pool if the system has multiple underlying drive types.
- Decide whether to create a capability profile for VMware VVols that has capabilities based on the pool configuration. To use the capability profile, you must assign specific usage tags, which are propagated to the VMware vSphere

environment, and can be used in policy profiles. The virtualization administrator and storage administrator should work together to define these tags.

Note

You cannot shrink a pool or change its storage characteristics without deleting the storage resources configured in the pool and the pool itself. However, you can add drives to expand the pool.

Procedure

1. Under **Storage**, select **Pools**.
2. Select the **Add** icon.
3. Select the tiers and virtual disks to use in the pool. Each virtual disk must have an assigned tier. If the virtual disks you want to include in the pool do not have assigned tiers, you must assign one. The tier you select for a virtual disk must be based on the underlying drive characteristics.
4. Optionally create a VMware capability profile for use by VVols, and specify usage tags for that profile.

About VMware host configurations

A host configuration defines a communication path through which a specific host or range of hosts can access storage resources. It also provides a mechanism by which you can manage access to storage resources by configuring the level of access permitted for particular host configurations.

Unisphere provides VMware discovery capabilities to collect virtual machine and datastore storage details from vSphere and display them in the context of the storage system. This automates the iSCSI target discovery for ESXi hosts to access the storage. In Unisphere, you can provision storage for a VMware datastore and configure access to the relevant ESXi host. The storage system then automatically connects to the ESXi host and configures the relevant datastore access. When you modify or delete a datastore in Unisphere, the storage system automatically updates the ESXi host to include the change or remove the datastore. If vCenter or ESXi host is created as a manual host, these automation tasks will not work. For VVol datastores, the Unity system must be registered as a VASA provider on the host to support this automation. You can register the Unity system as a VASA provider when adding host access in Unisphere.

Note

By default, the storage system automatically polls for updated configuration information every 24 hours. You can also choose to poll for updated configuration information at any time by selecting the polling options under **More Actions** of the appropriate VMware host tab.

vCenter server and ESXi host connections to VMware datastores

After you create a VMware datastore and configure access to it for a particular host configuration, you can connect the vCenter server or ESXi host to the storage resource using one of the following methods:

Table 1 Host access configuration methods

Datastore type	Method of connection
VMFS datastores	Use vSphere to re-scan for new storage devices. When the VMware datastore appears as an accessible storage device, add each VMFS datastore to the ESXi host.
NFS datastores	Use vSphere to add new network file system storage, specifying the following: • IP address of the associated NAS server • Export path to the datastore
VVol datastores	Hosts that have access to the respective NAS protocol endpoints or SCSI protocol endpoints will have access to the VVols File or VVols Block datastores that use these protocol endpoints.

Note

Automatic registration of the datastore in vCenter or the ESXi host is only available for automatically discovered hosts. For VVols datastores, the Unity system must be registered as a VASA provider on the host for automatic datastore registration in vCenter or vSphere. If you want to set up manual registration of a datastore on the ESXi host, you must manually register the host on the Unity system instead of using autodiscover.

Add a VMware vCenter server or ESXi host

Before you begin

Obtain the following information:

- Network name or IP address of the vCenter server or the ESXi host. Ensure that the vCenter server is available on the local network.
- User name and password of an account with access to the vCenter server.

Procedure

1. Under **Access**, select **VMware** > **vCenters**.
2. Select **Add**.
3. On the **Add vCenter** or **Add ESXi Host** window, enter the relevant details, and click **Find**.
4. From the list of discovered entries, select the relevant ESXi hosts, and click **Next**.
5. To register the Unity system as a VASA provider with the vCenter, select **Register VASA Provider** and enter the Unity Unisphere credentials.
6. On the Summary page, review the ESXi hosts, and click **Finish**.

Change ESXi host properties

Procedure

1. On the **General** tab, edit the description of the host.
2. On the **Network Addresses** tab, select an IP network address and click the **Edit** icon. Check the checkbox for any network addresses that should be ignored by the host. For example, you may want to ignore any network addresses used exclusively for system management.
3. On the **Initiators** tab:
   a. Select an FC initiator that you want the ESXi host to **Ignore**.

      Once an initiator is ignored, ESXi hosts will no longer be able to access any storage from it.

   b. Select an iSCSI initiator and select the **Edit** icon to change the CHAP properties.

Change vCenter properties

Procedure

1. Under **Access**, select **VMware** > **vCenters**.
2. Select a vCenter server and click the **Edit** icon.
3. Edit the description of the vCenter server.
4. Edit the credentials that the storage system uses to access the vCenter server.

Capability profiles

A VVol datastore is associated with one or more capability profiles. A capability profile is a set of storage capabilities for a VVol datastore. These capabilities are derived based on the underlying pools for the VVol datastore. The VVol datastore will show as compatible storage in vCenter or the vSphere Web Client if the associated capability profiles meet VMware storage policy requirements. Capability profiles must be created before you can create a VVol datastore. Capability profiles can be created at the time of pool creation (recommended), or can be added to an existing pool later.

You can define a capability profile in the following ways:

Table 2 Storage capabilities

Service level-based provisioning (physical deployments)	Expected service level for the pool:
	• Platinum
	  ▪ Single-tiered Flash pool
	• Gold
	  ▪ Multitiered pool with a mix of Flash and SAS drives
	  ▪ Single-tiered pools with SAS RAID 10
	• Silver

Table 2 Storage capabilities (continued)

	▪ Single-tiered pools with SAS RAID 5 or RAID 6 ▪ Multitiered pools with a mix of SAS and NL-SAS • Bronze ▪ Single-tiered pools with NL-SAS ▪ Multitiered pools with a mix of Flash and NL-SAS
Service level-based provisioning (virtual deployments)	Expected service level for a virtual pool: • Gold ▪ Multitiered pool with a mix of Extreme Performance and Performance tiers ▪ Single-tiered Extreme Performance pool • Silver ▪ Multitiered pool with a mix of Extreme Performance, Performance, and Capacity tiers ▪ Multitiered pool with a mix of Performance and Capacity tiers ▪ Single-tiered Performance pool • Bronze ▪ Multitiered pool with a mix of Extreme Performance and Capacity tiers ▪ Single-tiered Capacity pool
Usage tags	Usage tags can be applied to capability profiles to designate them and their associated VVol datastores for a particular use. For example, a VVol datastore may be tagged for VVols and VMs that support a particular application. The virtualization administrator and storage administrator should collaborate to define these usage tags.
Storage properties	Supported storage properties include: • Drive type: ▪ Extreme Performance [Flash] ▪ Performance [SAS] ▪ Capacity [NL-SAS] ▪ Multitier [mixed] ▪ Extreme Multitier [mixed with Flash] • RAID type (physical deployments only):

Table 2 Storage capabilities (continued)

	▪ RAID5 ▪ RAID6 ▪ RAID10 ▪ Mixed • FAST Cache (physical deployments only): ▪ Enabled ▪ Disabled • FAST VP tiering policy: ▪ Highest Available Tier ▪ Start High then Auto-Tier ▪ Auto-Tier ▪ Lowest Available Tier • Space Efficiency

Create a capability profile

Before you begin

Before creating a capability profile, you must create the pools that will be used by the VVol datastore.

It is recommended that you create capability profiles during pool creation. You can also add them to existing pools using the following method.

Note

You must create a capability profile before you can create a VVol datastore.

Procedure

1. Under **Storage**, select **VMware** > **Capability Profiles**.
2. Click the **Add** icon.
3. Enter a **Name** for the capability profile, and optionally add a **Description**.
4. Select the underlying **Pool** for the capability profile.
5. Optionally, enter any **Usage Tags** that will be used to identify how the associated VVol datastore should be used. For example, enter a particular application name or business unit that this datastore should be used for. The virtualization admin and the storage admin should work together to define usage tags.

Change a capability profile

Change an existing capability profile.

Procedure

1. Under **Storage**, select **VMware** > **Capability Profiles**.
2. Click the **Edit** icon.

3. On the **Details** tab, edit the **Name** and **Description**.
4. On the **Constraints** tab, edit the **Usage Tags**.
5. Click **Apply**.

Overview of configuring NAS servers

Before you can provision a VMware NFS datastore or file system storage, a NAS server that is appropriate for managing the storage type must be running on the system. A NAS Server is a file server that uses the SMB protocol, NFS protocol, or both to share data with network hosts. It also catalogs, organizes, and optimizes read and write operations to the associated file systems.

Configuring a NAS server requires specifying the following information:

- SP that the NAS server will run on.
- Pool used to store the NAS server's configuration data, such as anti-virus configurations, NDMP settings, network Interfaces and IP addresses.
- IP addresses that will be assigned to the NAS server to allow network hosts to access the shared data.

Performance balancing with NAS servers (physical deployments only)
You can balance the performance load on the storage system's SPs by choosing which NAS servers run on each SP, and which file systems are associated with which NAS server. For example, if you plan to provide file systems for two high-load database applications, you can choose to run a separate NAS server on each SP, and provision the storage for each application from a separate NAS server. This balances system performance by ensuring that the applications draw their processing resources from separate SPs.

File sharing with NAS servers
You can create NAS servers that support different types of file sharing. The table below describes the available NAS server configurations.

Table 3 NAS server configurations by operating environment

Operating Environment	NAS server function	Recommended configuration options
Unix-only environment	Provide only NFS access to file system data.	On the **Sharing Protocols** tab of the Create a NAS Server wizard, select **Linux/Unix shares (NFS)**.
Windows-only environment	Provide only SMB access to file system data.	On the **Sharing Protocols** tab of the Create a NAS Server wizard, select **Windows shares (SMB, CIFS)**.
Balanced Unix and Windows environment	Provide both SMB and NFS access to the same file systems data.	1. Make sure an NTP server is configured for the system. 2. Do the following in the Create a NAS Server wizard: • On the **Sharing Protocols** tab, select **Multiprotocol**. • Join the NAS server to a Windows Active Directory domain.

Table 3 NAS server configurations by operating environment (continued)

Operating Environment	NAS server function	Recommended configuration options
		• Configure local files, a Unix Directory Service (UDS), or both. If you configure local files with a UDS , the system queries the local files first. The UDS can be LDAP or NIS. • Configure DNS. 3. Optionally customize the mappings between Windows user accounts and Unix user accounts by modifying and uploading a user mapping file with advanced naming rules (ntxmap). You only need to do this when the names of the same users follow different naming rules in Windows and Unix.
Unix environment with the ability to access file system data through SMB	Provide NFS access to file system data and optionally provide SMB access to the same file system data for some user accounts.	1. Follow the steps in the Balanced Unix and Windows environment row for creating a NAS server and optionally customizing the mappings between Windows user accounts and Unix user accounts. 2. On the NAS server properties page for the new NAS server, select **Sharing Protocols** > **Multiprotocol**, and then configure a default Unix user account. All unmapped Windows accounts will be mapped to this user account. 3. When you create file systems for the NAS server, It is recommended that you specify a file system access policy of Unix.
Windows environment with the ability to access file system data through NFS	Provide SMB access to file system data and optionally provide NFS access to the same file system data for some user accounts.	1. Follow the steps in the Balanced Unix and Windows environment row for creating a NAS server and optionally customizing the mappings between Windows user accounts and Unix user accounts. 2. On the NAS server properties page for the new NAS server, select **Sharing Protocols** > **Multiprotocol**, and then optionally do either of the following: • Select **Enable automatic mapping for unmapped Windows accounts**. When you select this option, the system generates a Unix UID for each Windows users that is not already mapped to a Unix account through a directory service (LDAP or

Table 3 NAS server configurations by operating environment (continued)

Operating Environment	NAS server function	Recommended configuration options
		NIS) or local files. This functionality allows for the retention of file system quotas for each unmapped Windows user. (File system quotas are based on the Unix UID.) • Select **Enable default account for unmapped users** and configure a default Unix user account. All unmapped Windows accounts will be mapped to this Unix user account and will share the same file system quotas. 3. When you create file systems for the NAS server, It is recommended that you specify a file system access policy of Windows.

Create a NAS server for UNIX-only file sharing (NFS)

Before you begin

Obtain the following information:

- (Optional) Name of the tenant to associate with the NAS server.
- Name of the pool to store the NAS server's metadata.
- Storage Processor (SP) on which the NAS server will run.
- IP address information for the NAS server.
- VLAN ID, if the switch port supports VLAN tagging. If you associate a tenant with the NAS server, you must choose a VLAN ID.
- UNIX Directory Service (UDS) information for NIS or LDAP (optional). This can be used to resolve hosts defined on NFS share access lists.
- DNS server information (optional). This can also be used to resolve hosts defined on NFS share access lists.
- Replication information (optional).

It is recommended that you balance the number of NAS servers on both SPs.

Procedure

1. Under **Storage**, select **File** > **NAS Servers**.
2. Select the **Add** icon.
3. On the **General** and **Interface** pages, specify the relevant settings. Note the following:
   - On the **General** page, the **Server name** identifies the NAS server. It is not a network name.
   - Optionally select a tenant to associate with the NAS server.

Note

Once you create a NAS server that has an associated tenant, you cannot change this association.

- On the **Interface** page, optionally select a VLAN. If you selected a tenant on the **General** page, you must select a VLAN. The list of VLANs represent the VLANs associated with the selected tenant.

4. On the **Sharing Protocols** page:
   - Select **Linux/Unix shares (NFS)**.
   - Optionally enable support for Virtual Volumes (VVols) and NFSv4. Selecting **Enable NFSv4** enables support for both NFSv4 and NFSv3. The storage system supports NFSv3 by default.
   - Optionally click **Configure secure NFS** to enable secure NFS with Kerberos. When you enable secure NFS for a NAS server that supports UNIX-only file sharing, you must configure a custom Kerberos realm.
5. On the **Unix Directory Service** page, configure one of the following directory services (optional unless you are configuring secure NFS):
   - Local files
   - NIS
   - LDAP
   - Local files and NIS or LDAP

   If you configure local files with NIS or LDAP , the system queries the local files first. You can configure LDAP to use anonymous, simple, and Kerberos authentication. You can also configure LDAP with SSL (LDAP Secure) and can enforce the use of a Certificate Authority certificate for authentication.
6. On the **DNS** page, optionally configure DNS for the NAS server.
7. On the **Replication** page, optionally select a replication mode and Recovery Point Objective (RPO) for the NAS server.

Create a NAS server for multiprotocol file sharing (SMB and NFS)

Before you begin

When you create a NAS server that supports multiprotocol file sharing, it must be joined to an Active Directory (AD). This requires that an NTP server is configured on the storage system.

Obtain the following information:

- (Optional) Name of the tenant to associate with the NAS server.
- Name of the pool to store the NAS server's metadata.
- Storage Processor (SP) on which the NAS server will run.
- IP address information for the NAS server.
- VLAN ID, if the switch port supports VLAN tagging. If you associate a tenant with the NAS server, you must choose a VLAN ID.
- AD information, including the SMB computer name (used to access SMB shares), and either the domain administrator's credentials or the credentials of a user of the domain who has privileges for joining the AD. You can optionally specify the NetBIOS name and organizational unit. The NetBIOS name defaults to the first 15

characters of the SMB server name. The organizational unit defaults to OU=Computers,OU=EMC NAS servers.

- UNIX Directory Service (UDS) information for NIS, LDAP, or local files. The UDS provides the UNIX UID and GUID for AD users.

  Note

  You can configure mappings for some users in the UDS and let the others be mapped through the default account.

- DNS server and domain information.
- Replication information (optional).

It is recommended that you balance the number of NAS servers on both SPs.

You cannot disable multiprotocol file sharing for a NAS server once a file system is created on that NAS server.

Procedure

1. Under **Storage**, select **File** > **NAS Servers**.
2. Select the **Add** icon.
3. On the **General** and **Interface** pages, specify the relevant settings while noting the following:
   - On the **General** page, the **Server name** identifies the NAS server. It is not a network name.
   - Optionally select a tenant to associate with the NAS server.

     Note

     Once you create a NAS server that has an associated tenant, you cannot change this association.

   - On the **Interface** page, optionally select a VLAN. If you selected a tenant on the **General** page, you must select a VLAN. The list of VLANs represent the VLANs associated with the selected tenant.
4. On the **Sharing Protocols** page:
   - Select **Multiprotocol**, and join the NAS server to the AD.
   - Optionally click **Advanced** to change the default NetBios name and organizational unit.
   - Optionally enable support for Virtual Volumes (VVols) and NFSv4. Selecting **Enable NFSv4** enables support for both NFSv4 and NFSv3. If you select **Linux/Unix shares (NFS)** without selecting **Enable NFSv4**, the storage system supports NFSv3 only.
   - Optionally click **Configure secure NFS** to enable secure NFS with Kerberos. When you enable secure NFS, you can choose to authenticate using the Windows Kerberos realm (that is, the Windows domain) configured on the NAS server, or you can configure and use a custom realm.

     Note

     It is recommended that you use LDAPS with secure NFS.

5. On the **Unix Directory Service** page, configure one of the following directory services:

- Local files
- NIS
- LDAP
- Local files and NIS or LDAP

If you configure local files with NIS or LDAP, the system queries the local files first. You can configure LDAP to use anonymous, simple, and Kerberos authentication. You can also configure LDAP with SSL (LDAP Secure) and can enforce the use of a Certificate Authority certificate for authentication.

6. On the **DNS** page, configure DNS for the NAS server.
7. On the **Replication** page, optionally select a replication mode and Recovery Point Objective (RPO) for the NAS server.

Change NAS server properties

The following rules apply to changing NAS server settings:

- You cannot disable multiprotocol file sharing for a NAS server once a file system is created on that NAS server.
- If you disable multiprotocol file sharing on a NAS server, the NAS server still has the NFS and SMB protocols enabled, but no longer supports simultaneously sharing NFS and SMB file systems.
- You cannot disable DNS for:
  - NAS servers that support multiprotocol file sharing.
  - NAS servers that support SMB file sharing and that are joined to an Active Directory (AD).
- To reconfigure a NAS server that supports SMB-only or NFS-only file systems so that it supports multiprotocol (both types of file systems simultaneously), first enable a UNIX Directory Service and DNS server for that NAS server.

Procedure

1. Under **Storage**, select **File** > **NAS Servers**.
2. Select the relevant NAS server, and then select the **Edit** icon.
3. On the **General** tab:
   - Change the NAS server name.
   - Select **SP Owner** to transition from one SP to the other SP for this NAS server. For example, you may want to do this if you have an overloaded SP, and want to reduce the load by moving the server to the other SP.
4. On the **Network** tab:
   - Select the **Interfaces & Routes** sub-tab to add, change, delete, or verify NAS server interfaces, enable or disable IP packet reflect for the NAS server, or change the NAS server's preferred interfaces. Select an interface, and then select **Show external routes for interfaces** to access the per-interface routing table, where you can add, change, or delete the selected interface's routes for responding to client requests.
   - Select the **Routes to External Services** sub-tab to add, change, or verify NAS server routes for external service requests, or to configure default gateways.

5. On the **Naming Services** tab, configure DNS and either configure the UNIX Directory Service (UDS) for the NAS server (LDAP or NIS) or use local files. Alternatively, you can use local files with a UDS. In this case, the system checks the local files first.
6. On the **Sharing Protocols** tab:
   - Select the **SMB** sub-tab to enable or disable support for Windows shares and to change SMB properties.
   - Select the **NFS** sub-tab to enable or disable support for NFS shares, VVols, NFSv4, and extended UNIX credentials. You can also configure secure NFS with Kerberos and change the credential cache retention period.
   - Select the **FTP** sub-tab to enable or disable FTP and to change FTP properties.
   - Select the **Multiprotocol** sub-tab to enable or disable multiprotocol file sharing and to specify default Windows and UNIX accounts for unmapped users. You can also work with user mapping files, run user mapping diagnostics, and have the storage system automatically update user mappings on all file systems.
7. On the **Protection & Events** tab:
   - Select the **NDMP Backup** sub-tab to enable or disable NDMP, and to change the NDMP password.
   - Select the **DHSM** sub-tab to enable or disable Distributed Hierarchical Storage Management (DHSM) and to change the DHSM password.
   - Select the **Events Publishing** sub-tab to enable or disable Events Publishing, create or modify an event pool, and create or modify events policy settings.
8. On the **Security** tab:
   - Select the **Antivirus** sub-tab to enable or disable the antivirus service and to retrieve or upload the antivirus configuration file.
   - Select the **Kerberos** sub-tab to configure a custom Kerberos realm and to retrieve or upload the Kerberos keytab file.
9. On the **Replication** tab, optionally select a replication mode and Recovery Point Objective (RPO) for the NAS server.

Protocol endpoints

Protocol Endpoints (PEs) are access points for ESXi host communication to the storage system. These endpoints establish a datapath on-demand for virtual machines and their respective VVol datastores. I/O from VMs is communicated through the PE to the VVol datastore on the storage system. A single protocol endpoint can multiplex I/O requests from a large number of VM clients to their virtual volumes. Protocol endpoints are automatically created when a host is granted access to a VVol datastore.

NAS protocol endpoints are created and managed on the storage system and correspond to a specific NFS-based NAS server. A File VVol will be bound to the associated NAS PE every time that VM is powered on. When the VM is powered off, the VVol is unbound from the PE.

SCSI protocol endpoints can utilize any iSCSI interface or Fibre Channel connection for IO. Two iSCSI PEs are created for every ESXi host-to-VVol datastore (storage

container) pair; this ensures high-availability. The Block VVol will be bound to the associated SCSI PE every time that the VM is powered on. When the VM is powered off, the PE is unbound. SCSI protocol endpoints are like LUN mount points that allow I/O access to VVols from the ESXi host to the storage system.

NAS protocol endpoint servers

VMware protocol endpoint servers are NFS-based NAS servers enabled to provide an I/O path from the VMware host to it's respective File VVol datastore on the storage system.

You can enable a NAS server for VVols in the **Create a NAS server** wizard. The IP address assigned to the NAS server at creation time becomes the **Advertised IP address** for the NAS protocol endpoint. When enabling VVols on an existing NAS server, you can select which IP address should be the **Advertised IP address** from the list of IP interfaces already created for the NAS server. It is recommended that you enable at least two NAS servers for VVols, one on each SP, for high availability. The system will select one of these NAS PEs automatically based on which will maximize throughput.

Change VMware protocol endpoint information

Procedure

1. Under **Storage**, select **VMware** > **Protocol Endpoints**.
2. Click the **Edit** icon.
3. On the **General** tab, edit the description of the protocol endpoint.
4. On the **Host Access** tab, change your selections of which hosts have access to use the protocol endpoint.

VVol datastores

VVols reside in VVol datastores, also known as storage containers, which are comprised of storage allocations from one or more capability profiles. Capability profiles are built on top of one or more underlying pools. You can create VVol datastores based on one or more capability profiles and then allocate a specific amount of space from the capability profile to the VVol datastore.

Each VVol datastore has one or more capability profiles that describe its performance and capacity characteristics, such as drive type, FAST VP tiering policy, and space efficiency policy. These characteristics are derived based on the underlying pool. When a virtual volume is created in vSphere, it is assigned a storage policy profile. vSphere filters the compatible and incompatible available VVol datastores (from one or more storage systems) when the VVol is being created based on these profiles. Only VVol datastores that support the storage policy profile are considered compatible storage containers for deploying the VVol.

Create a VMware VVol datastore

Before you begin

You must create capability profiles before creating a VVol datastore.

Procedure

1. Under **Storage**, select **VMware** > **Datastores**.
2. Click the **Add** icon.
3. On the **Type** page, select **VVOL (File)** or **VVOL (Block)**.
4. Enter a **Name** and optionally a **Description** for the VVol datastore.
5. Select one or more capability profiles that will be used by the VVols datastore.
   a. Optionally, click on the current size or **Edit** in the **Datastore Size (GB)** column to adjust the space allocated from the pool to each selected capability profile.
   b. Adjust the size and/or unit of measure (TBs or GBs) of the capability profile.
   c. Click **OK**.
6. Select the hosts that will have **Access** to the datastore.

Change a VVol datastore

Procedure

1. Under **Storage**, select **VMware** > **Datastores**.
2. Select the datastore and click the **Edit** icon.
3. On the **General** tab, edit the **Name** and **Description**. Click **Apply**.
4. On the **Capability Profiles** tab, edit the selected capability profiles used for the VVol datastore.

   To change the size of an existing capability profile:

   a. Click on the current size in the **Datastore Size (GB)** column for the capability profile.
   b. Adjust the size and/or unit of the capability profile.
   c. Click **OK**.

   To add a new capability profile:

   a. Click **Add** to add a new capability profile to the VVol datastore.

      This will open a new window with the list of available capability profiles on the system.

   b. Select a new capability profile for the VVol datastore and click **OK**.

   To delete an existing capability profile not currently in use:

   a. Select the capability profile.
   b. Click the **Delete** icon.
5. On the **Host Access** tab, edit the hosts that have access to the datastore.

Types of VVol objects

Virtual volumes are encapsulations of virtual machine files, virtual disks, and their derivatives. There are several types of VVol objects that correspond to an individual virtual volume, including a VMDK VVol (data VVol), Config VVol, Memory VVol, and Swap VVol.

Table 4 Types of VVols

VMDK (Data) VVol	The VMDK VVol, displayed as **Data** VVol in Unisphere, contains the vDisk file, or the hard disk drive, for the VM.
Config VVol	The **Config** VVol contains settings, configuration, and state information for the VM. This includes .vmx, nvram, and log files.
Memory VVol	The **Memory** VVol contains a complete copy of the VM memory as part of a with-memory VM snapshot.
Swap VVol	The **Swap** VVol is created when VMs are powered on and contain copies of the VM memory pages that are not retained in memory.

About VASA support

The VMware vSphere APIs for Storage Awareness (VASA) is a set of APIs that provides storage awareness to VMware vSphere clients. It enables vSphere clients to request and display basic information on the storage system and the storage resources it exposes to the virtual environment. Using the VASA protocol, you can configure the vSphere client to view information on physical storage system objects that are associated with the storage system datastores. This information includes storage policies and properties, such as tiering and RAID level. You can also view the health status of these components in vSphere. Changes in the health status or information about storage resources reaching space capacity thresholds are reported as VASA alarms in the vSphere client.

VASA has introduced new APIs to support virtual volumes (VVols) starting with vSphere 6.0. These updated VASA APIs enhance storage system awareness of individual VM disks. This enables the storage system to perform operations on individual VM disks such as snapshots and clones.

Note

The Unity system can be registered as a VASA provider automatically in vSphere when corresponding vCenter and ESXi hosts are created, and the option to add Unity as a VASA provider is enabled. Unity can only be registered as a VASA provider for one vCenter server at a time. Refer to the Unity *Configuring VVols* guide for additional options.

Add the system as a VASA provider

Note

The Unity system is registered as a VASA provider automatically in vSphere when corresponding vCenter and ESXi hosts are granted access to the system.

For the vCenter server to communicate with the system, add the system as a storage provider in the vSphere client. Use the following information:

- **Name** - Name of the storage provider that will appear in the vSphere client. You can choose to use any name you want.
- **URL** - The VASA Provider service URL. The URL must be in the following format: https://<management IP address>: 8443/vasa/version.xml
- **Login** - Unisphere user name with the Administrator or VM Administrator role. It is recommended that you specify a user account with the VM Administrator role. Note the following syntax:
  - For local users: `local/<user name>`
  - For LDAP users: `<domain>/<user name>`
- **Password** - The password associated with the user account.

For more information on adding a storage provider, refer to the VMware documentation.

Note

If you create VM Storage Policies in vSphere during the same vSphere login session where you added the storage system as a VASA provider, rule set labels may appear as ID strings instead of the correct rule set names. Logging out of vSphere and logging back in may resolve this issue.

CHAPTER 2

Manage VMware virtual volume datastores with CLI

This chapter addresses the following topics:

Create a NAS server

Create a NAS server.

Note

The NFSv3 protocol is enabled by default when creating a NAS server.

Format

```
/net/nas/server create -name <value> -sp <value> {-pool <value>
| -poolName <value>} [-tenant <value>] [-mpSharingEnabled {no |
yes [-autoUserMappingEnabled {yes | no}][-unixDirectoryService
{local | ldap | nis | localThenNis | localThenLdap | none}] [-
defaultUnixUser <value>] [-defaultWindowsUser <value>]}] [-
replDest {yes | no}] [-enablePacketReflect {yes | no}]
```

Action qualifiers

Qualifier	Description
`-name`	Specifies the NAS server name. **Note** NAS server names can contain alphanumeric characters, a single dash, and a single underscore. Server names cannot contain spaces or begin or end with a dash. You can create NAS server names in four parts that are separated by periods (example: aa.bb.cc.dd). Names can contain up to 255 characters, but the first part of the name (before the first period) is limited to 15 characters.
`-sp`	Specifies the parent SP for the NAS server. Value is SPA or SPB.
`-pool`	Specifies the ID of the storage pool for the NAS server.
`-poolName`	Specifies the name of the storage pool for the NAS server.
`-tenant`	Specifies the tenant identifier. **Note** If a tenant is not specified, the NAS server is created in the default network namespace.
`-mpSharingEnabled`	Indicates whether multiprotocol sharing mode is enabled. Value is yes or no (default).
`-unixDirectoryService`	Directory Service used for querying identity information for Unix (such as UIDs, GIDs, net groups). Valid values are: • `nis`

Qualifier	Description
	• ldap • local • none (default) • localThenNis • localThenLdap
-autoUserMappingEnabled	Indicates whether a Windows user who is not mapped to a known Unix/Linux username is allowed to access the NAS server's files Valid values are: • yes— The system generates an internal UID for the Windows user and allows access to the NAS server's files through Windows. • no (default)— The Windows authentication fails unless there is a default Unix username configured.
-defaultUnixUser	Default Unix user name or Unix ID that grants file access in the multiprotocol sharing mode. This user name or ID is used when the corresponding Unix/ Linux user name or ID is not found by the mapping mechanism. The Unix ID format is @uid=xxxx,gid=yyyy@, where xxxx and yyyy are the decimal numerical values of the UID and the primary GID, respectively. When using this ID, the user does not need to be defined in the UDS.
-defaultWindowsUser	Default Windows user name that grants file access in the multiprotocol sharing mode. This user name is used when the corresponding Windows user name is not found by the mapping mechanism.
-replDest	Replication destination settings for the NAS server. When this option is set to yes, only mandatory parameters may be included. All other optional parameters will be inherited from the source NAS server. Valid values are: • yes • no (default)
-enablePacketReflect	Indicates whether the reflection of outbound (reply) packets through the same interface that inbound (request) packets entered is enabled. Valid values are: • yes (default) • no

Example

The following command creates a NAS server with these settings:

- Name is NasServer_1.
- Associated with SP A.
- Associated with storage pool pool_0.
- IP Packet Reflect is enabled.
- The ID of the new NAS server is ID nas_1.

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /net/nas/server
create -name NasServer_1 -sp spa -pool pool_0 -enablePacketReflect yes
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

ID = nas_1
Operation completed successfully.
```

Change NAS server settings

Modify an existing NAS server.

Format

```
/net/nas/server {-id <value | -name <value } set [-name
<value>] [-sp {spa | spb}] [-mpSharingEnabled {yes | no}] [-
unixDirectoryService {ldap | nis | none}] [-
autoUserMappingEnabled {yes | no}] [{-defaultAccessDisabled |
[-defaultUnixUser <value>] [-defaultWindowsUser <value>]}] [-
enablePacketReflect {yes | no }] [-replDest {yes | no }] [-
preferredProductionOverride { no | yes }][-
preferredProductionIPv4 { auto | <value>}] [-
preferredProductionIPv6 { auto | <value>}] [-
preferredBackupIPv4 {auto | <value>}] [-preferredBackupIPv6
{auto | <value>}
```

Object qualifiers

Qualifier	Description
-id	Type the ID of the NAS server to change.
-name	Type the name of the NAS server to change.

Action qualifiers

Qualifier	Description
-name	Shared folder server name.
-sp	Owner SP. Valid values are: • spa • spb
-mpSharingEnabled	Indicates whether multiprotocol sharing mode is enabled. Valid values are: • yes • no

Qualifier	Description
	Note You cannot disable multiprotocol file sharing for a NAS server once a file system is created on that NAS server.
`-unixDirectoryService`	Directory Service used for querying identity information for Unix (such as UIDs, GIDs, net groups). Valid values are: • `nis` • `ldap`
`-defaultAccessDisabled`	Disables file access when no user mapping mechanism is found.
`-autoUserMappingEnabled`	Indicates whether a Windows user who is not mapped to a known Unix/Linux username is allowed to access the NAS server's files Valid values are: • `yes`. The system generates an internal UID for the Windows user and allows access to the NAS server's files through Windows. • `no` (default). The Windows authentication fails unless there is a default Unix username configured.
`-defaultUnixUser`	Default Unix user name or Unix ID that grants file access in the multiprotocol sharing mode. This user name or ID is used when the corresponding Unix/Linux user name or ID is not found by the mapping mechanism. The Unix ID format is @uid=xxxx,gid=yyyy@, where xxxx and yyyy are the decimal numerical values of the UID and the primary GID, respectively. When using this ID, the user does not need to be defined in the UDS.
`-defaultWindowsUser`	Default Windows user name that grants file access in the multiprotocol sharing mode. This user name is used when the corresponding Windows user `-defaultWindowsUser` name is not found by the mapping mechanism.
`-replDest`	Replication destination settings for the NAS server. Valid values are: • `yes` • `no`

Qualifier	Description
-enablePacketReflect	Indicates whether the reflection of outbound (reply) packets through the same interface that inbound (request) packets entered is enabled. Valid values are: • yes • no
-preferredProductionOverride	Override the replicated production interfaces "preferred interface" settings. Valid values are: • yes • no
-preferredProductionIPv4	Production IPv4 preferred interface settings. The interface must be IPv4 and belong to this server. Valid values are: • *<interface ID>* • auto
-preferredProductionIPv6	Production IPv6 preferred interface settings. The interface must be IPv6 and belong to this server. Valid values are: • *<interface ID>* • auto
-preferredBackupIPv4	Backup and DR test IPv4 preferred interface settings. The interface must be IPv4 and belong to this server. Valid values are: • *<interface ID>* • auto
-preferredBackupIPv6	Backup and DR test IPv6 preferred interface settings. The interface must be IPv6 and belong to this server. Valid values are: • *<interface ID>* • auto

Example 1
The following command updates NAS server nas_1 with these settings:

- Enables multiprotocol sharing.
- Uses LDAP as the Unix Directory Service.

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /net/nas/server -id
nas_1 set -mpSharingEnabled yes -unixDirectoryService ldap
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection
```

```
ID = nas_1
Operation completed successfully.
```

Example 2

The following command changes the replication settings for NAS server nas_1.

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /net/nas/server -id
nas_1 set -replDest yes
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

ID = nas_1
Operation completed successfully.
```

Example 3

The following command changes the storage processor to SPB for NAS server nas_1.

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /net/nas/server -id
nas_1 set -sp spb
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

WARNING: Modifying the NAS server's SP disrupts any running NDMP
jobs, and may also result in data unavailability for some client
configurations other than NFS (v3, v4, and v4.1) and SMB3+CA. The
NDMP jobs must be restarted after the SP modification is completed.
Are you sure you want to modify the default SP?
yes / no:yes

ID = nas_1
Operation completed successfully.
```

Note

- When the SP is being modified, the NAS server health attribute is updated to INFO, and the health details attribute is updated to `Transitioning to other Storage Processor`. When the SP modification completes, the NAS server health and health details are reverted back to the previous values.
- A change to the SP cannot be performed on a NAS Server that is part of an active VDM File Import operation. The Import operation must be completed before the SP can be changed. Otherwise, the following error occurs: `Failed: Cannot complete the operation because the resource is under import. (Error Code:0x900012a).`
- A change to the SP cannot be performed on a NAS Server that is part of an active replication session. Pause the replication session, perform the SP change, and then resume the replication session. Otherwise, the following error occurs: `Cannot modify the NAS server's Storage Processor when there are non-paused replication sessions on the NAS server or its file systems. (Error Code:0x6720665).`

Manage VMware NAS protocol endpoint servers

VMware protocol endpoint servers are NFS-based NAS servers enabled to provide an I/O path from the VMware host to it's respective File VVol datastore on the storage system.

When creating a NAS protocol endpoint server, you can choose which IP address the NAS PE will use from the list of IP interfaces already created for the NAS server. It is recommended that you enable at least two NAS servers for VVols, one on each SP, for high availability. The system will select one of these NAS PEs automatically based on which will maximize throughput.

Table 5 Protocol endpoint server attributes

Attribute	Description
`ID`	VMware protocol endpoint identifier.
`NAS server`	Identifier of the associated NAS server for NAS PEs.
`NAS server interface`	Identifier of the NAS server IP interface to be used by the VMware NAS protocol endpoint server.

Note

Only one VMware protocol endpoint server per NAS server is supported.

Create protocol endpoint servers

Create VMware protocol endpoints servers for File VVols.

Format

```
/net/nas/vmwarepe create [-async] {-server <value> | -
serverName <value>} -if <value>
```

Action qualifier

Qualifier	Description
-async	Run the operation in asynchronous mode.
-server	Type the identifier of the NAS server.
-serverName	Type the name of the NAS server.
-if	Type the name of the identifier for the NAS IP interface to be used by the VMware protocol endpoint server.

Example

The following example creates a protocol endpoint server on NAS server "nas_1" with the IP interface "if_1".

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /net/nas/vmwarepe
create -server nas_1 -if if_ 1
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

ID = PES_0
Operation completed successfully.
```

View VMware protocol endpoint servers

View VMware protocol endpoints servers for File VVols.

Format

```
/net/nas/vmwarepe [{-id <value> | -server <value> | -serverName
<value>}] show
```

Action qualifier

Qualifier	Description
-id	Type the identifier of the NAS protocol endpoint server.
-server	Type the identifier of the associated NAS server.
-serverName	Type the name of the associated NAS server.

Example

The following example shows the details for all of the VMware protocol endpoint servers on the system.

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456!/net/nas/vmwarepe
show -detail
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection
```

```
1:     ID                    = PES_0
       NAS server            = nas_1
       NAS server interface = if_1
```

Delete protocol endpoint servers

Delete a VMware protocol endpoints server.

Format

`/net/nas/vmwarepe -id <value> delete [-async] [-force]`

Object qualifiers

Qualifier	Description
`-id`	Type the identifier or the VMware protocol endpoint server to be deleted.

Action qualifiers

Qualifier	Description
`-async`	Run the operation in asynchronous mode.
`-force`	Unconditionally removes all VMware NAS protocol endpoints using the VMware protocol endpoint server and unbinds all virtual volumes using the protocol endpoint server.

Example

The following example deletes VMware NAS protocol endpoint server "PES_0".

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /net/nas/vmwarepe -
id PES_0 delete
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

Operation completed successfully.
```

Manage host configurations

Hosts are the clients or servers in your network that access storage on the system. Host configurations are logical connections through which hosts or applications can access storage resources. Before a host can access storage, you must define a configuration for it and associate it with a storage resource. Create a host configuration for each host, host subnetwork (subnet), or network group (netgroup) that will access storage resources on the system.

You can create the following types of host configurations:

- Individual host configurations — Enable you to define and control access to storage resources on a host-by-host basis.
- Subnet and netgroup configurations — Enable you to define and control access to storage resources for multiple hosts or network segments.

Each host configuration is identified by an ID.

The following table lists the attributes for host configurations.

Table 6 Host configuration attributes

Attribute	Description
`ID`	ID of the host configuration.
`Name`	Name of the host configuration.
`Description`	Brief description of the host configuration.
`Tenant`	Tenant with which the host is associated.
`Address`	Hostname or IP address associated with the host, IP address of the subnet, or name of the netgroup. **Note** This information is required when connecting hosts to network shares on the system.
`Netmask`	Subnet mask for the host.
`Type`	Type of host configuration. Value is one of the following: • `host` — A host defines and controls access to storage resources on a host-by-host basis. • `subnet` — A subnet is a logical grouping of connected network devices. Devices on a subnet share contiguous ranges of IP addresses. A subnet mask, or network mask, defines the boundaries of an IP subnet. You can associate a host configuration with a subnet mask to define and control storage access for hosts on a particular network segment. • `netgroup` — A netgroup is a named sets of hosts, users, or domains on a network. A netgroup can provide a way to reference sets of Linux/UNIX hosts collectively for accessing storage over NFS. You can create a host configuration for a netgroup to define and control storage access for multiple Linux/UNIX hosts or users through a single configuration.
`OS type`	Type of operating system (OS) running on the host. You can enter any value you want. Here are suggestions for some of the common operating systems: • `undefined` — OS is not specified (default) or unknown. • `other` — Other. • `win2003srv` — Windows Server 2003. • `winxp` — Windows XP. • `win2008srv` — Windows Server 2008. • `winvista` — Windows Vista. • `win2012srv` — Windows Server 2012. • `esx` — VMware ESX. • `redhat` — Red Hat Enterprise Linux.

Table 6 Host configuration attributes (continued)

Attribute	Description
	• `sles` — SUSE Linux Enterprise. • `win7` — Windows 7. • `hyperv` — Microsoft Hyper-V. • `solaris` — Solaris.
`Ignored address`	A comma-separated list of host IP addresses to exclude from data access.
`Health state`	Health state of the host. The health state code appears in parentheses. Value is one of the following: • `Unknown (0)` — Status is unknown. • `OK (5)` — Working correctly. • `OK BUT (7)` — Working correctly, but there could be a problem. • `Degraded/Warning (10)` — Working and performing all functions, but the performance may not be optimum. • `Minor failure (15)` — Working and performing all functions but overall performance is degraded. This condition has a minor impact on the system and should be remedied at some point, but does not have to be fixed immediately. • `Major failure (20)` — Failing and some or all functions may be degraded or not working. This condition has a significant impact on the system and should be remedied immediately. • `Critical failure (25)` — Failed and recovery may not be possible. This condition has resulted in data loss and should be remedied immediately. • `Non-recoverable error (30)` — Completely failed and cannot be recovered.
`Health details`	Additional health information. See Appendix A, Reference, for health information details.
`Management type`	Indicates the way the host is managed. Value is one of the following: • `VMware` — The host is managed through VMware web services. • `Other` — The host is automatically created on the storage system. • `Manual` — The host is created manually.
`Accessible LUNs`	A comma-separate list of LUNs that are accessible to the host.

Create host configurations

Create a host configuration to establish a connection between the system and hosts that access the system.

Format

```
/remote/host create -name <value> [-descr <value>] [-tenant
<value>] -type {host [-addr <value>] [-ignoredAddr <value>] [-
osType <value> ] | subnet -addr <value> [-netmask <value>] |
netgroup -addr <value>}
```

Action qualifier

Qualifier	Description
`-name`	Specifies the name of the host configuration.
`-descr`	Specifies a brief description of the host configuration.
`-type`	Specifies the type of host configuration. Value is one of the following: • `host` — A host defines and controls access to storage resources on a host-by-host basis. • `subnet` — A subnet is a logical grouping of connected network devices. Devices on a subnet share contiguous ranges of IP addresses. A subnet mask, or network mask, defines the boundaries of an IP subnet. You can associate a host configuration with a subnet mask to define and control storage access for hosts on a particular network segment. • `netgroup` — A netgroup is a named sets of hosts, users, or domains on a network. A netgroup can provide a way to reference sets of Linux/UNIX hosts collectively for accessing storage over NFS. You can create a host configuration for a netgroup to define and control storage access for multiple Linux/UNIX hosts or users through a single configuration.
`-tenant`	Specifies the identifier of the tenant with which the host is to be associated. **Note** If not specified, the host is created in the default network namespace and the tenant attribute will be blank.
`-addr`	Specifies the hostnames or IP addresses associated with the host, IP addresses of the subnet, or the name of the netgroup. Separate each value with a comma. • Format: *<IP address>*/[*<prefix length>*]. • Default prefix length for IPv4 addresses is 24 and for IPv6 addresses is 64. **Note** This information is required when connecting hosts to network shares on the system.
`-ignoredAddr`	Specifies a list of IP addresses associated with the host that are excluded from data access. Separate each value with a comma.

Qualifier	Description
-netmask	Specifies the subnet mask for the host configuration.
-osType	Specify the type of operating system (OS) running on the host. You can enter any value you want. Here are suggestions for some of the common operating systems: • undefined — OS is not specified (default) or unknown. • other — Other. • win2003srv — Windows Server 2003. • winxp — Windows XP. • win2008srv — Windows Server 2008. • winvista — Windows Vista. • win2012srv — Windows Server 2012. • esx — VMware ESX. • redhat — Red Hat Enterprise Linux. • sles — SUSE Linux Enterprise. • win7 — Windows 7. • hyperv — Microsoft Hyper-V. • solaris — Solaris.

Example 1

The following command creates a host configuration for a host with these settings:

- Name is MyHost.
- Description is “accounting”.
- IP address is 10.64.74.10.
- OS is Windows XP.

The host configuration receives ID Host_1014:

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /remote/host create
-name MyHost -descr "accounting" -type host -addr 10.64.74.10 -osType
winxp
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

ID = Host_1014
Operation completed successfully.
```

Example 2

The following command creates a host configuration for a subnet with these settings:

- Name is MySubnet.
- Description is “subnet1”.
- IP address is 192.168.10.0.
- Subnet mask is 255.255.255.0.

The host configuration receives ID Subnet_1015:

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /remote/host create
-name MySubnet -descr "subnet1" -type subnet -addr 192.168.10.0 -
netmask 255.255.255.0
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

ID = Subnet_1015
Operation completed successfully.
```

Example 3

The following command creates a host configuration for a subnet with these settings:

- Name is IPv6Subnet.
- Description is "V6_HE_Subnet".
- IPv6 address is 2001:db8:c25:
- Prefix length is 48.

The host configuration receives ID NetGroup_1023:

```
uemcli -d 10.0.0.1 /remote/host create -name IPv6Subnet -descr
"V6_HE_Subnet" -type subnet -addr 2001:db8:c25::/48
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

ID = NetGroup_1023
Operation completed successfully.
```

View host configurations

View details about a host configuration. You can select the ID of the host configuration or the host type.

Format

```
/remote/host [{{-id <value> | -name <value>} | -type {host |
subnet | netgroup}}] show
```

Object qualifier

Qualifier	Description
-id	Specify the host ID.
-name	Specify the host name.
-type	Specifies the host type. Valid values are: • host • subnet • netgroup

Example

The following command lists all host configurations on the system:

uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /remote/host show -brief

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

1:     ID              = 1014
       Name            = MyHost
       Description     = this is my host
       Tenant          = tenant_3
       Type            = host
       Address         = 10.64.74.10, 10.64.80.10
       Netmask         =
       OS type         = winxp
       Ignored address = 10.64.80.10
       Health state    = OK (5)

2:     ID              = 1015
       Name            = MySubnet
       Description     = this is my subnet
       Tenant          =
       Type            = subnet
       Address         = 192.168.10.0
       Netmask         = 255.255.255.0
       OS type         =
       Ignored address =
       Health state    = OK (5)
```

Change host configuration settings

Change the settings for a host configuration.

Format

`/remote/host {-id <value> | -name <value>} set [-name <value>] [-descr <value>] [-addr <value>] [-ignoredAddr <value>] [-netmask <value>] [-osType <value>] [-addLuns <value>] [-removeLuns <value>]`

Object qualifier

Qualifier	Description
`-id`	ID of the host configuration to change.
`-name`	Name of the host configuration to change.

Action qualifier

Qualifier	Description
`-name`	Specifies the new name for the host configuration.
`-desc`	Specifies the new description of the host configuration.
`-addr`	Specifies the hostnames or IP addresses associated with the host, IP addresses of the subnet, or the network addresses of the netgroup. Separate each value with a comma. • For subnet type, specifies the new IP address of the subnet. • For netgroup, specifies the new netgroup's name.

Qualifier	Description
	• Format: *<IP address>/*[*<prefix length>*]. • Default prefix length for IPv4 addresses is 24 and for IPv6 addresses is 64. **Note** This information is required when connecting hosts to network shares on the system.
`-ignoredAddr`	Specifies a list of IP addresses associated with the host that are excluded from data access. Separate each value with a comma.
`-netmask`	Specify the subnet mask for the host configuration.
`-osType`	Specify the type of operating system (OS) running on the host. You can enter any value you want. Here are suggestions for some of the common operating systems: • `undefined` — OS is not specified or unknown. • `other` — Other. • `win2003srv` — Windows Server 2003. • `winxp` — Windows XP. • `win2008srv` — Windows Server 2008. • `winvista` — Windows Vista. • `win2012srv` — Windows Server 2012. • `esx` — VMware ESX. • `redhat` — Red Hat Enterprise Linux. • `sles` — SUSE Linux Enterprise. • `win7` — Windows 7. • `hyperv` — Microsoft Hyper-V. • `solaris` — Solaris.
`-addLuns`	Specify a comma-separated list of LUN friendly IDs for LUNs to add to the host.
`-removeLuns`	Specify a comma-separated list of LUN friendly IDs for LUNs to remove from the host.

Example

The following command updates the description of host configuration 1014 to indicate that it now holds the payroll database:

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /remote/host -id
1014 set -descr "Accounting" -osType winxp
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection
```

```
ID = 1014
Operation completed successfully.
```

Delete host configurations

Delete a host configuration.

NOTICE

Deleting a host configuration breaks the block-based (Fibre Channel or iSCSI) storage connections associated with the configuration. Hosts that use the configuration for NFS-based storage connections, such as NFS shares, revert to the default access privileges for any storage resources that they can access.

Format

`/remote/host {-id <value> | -name <value>} delete`

Object qualifier

Qualifier	Description
`-id`	ID of the host configuration to delete.
`-name`	Name of the host configuration to delete.

Example

The following command deletes host configuration 1014:

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /remote/host -id
1014 delete
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

Operation completed successfully.
```

Manage host initiators

After you create a host configuration for controlling host access to storage on the system, you need to create one or more initiators for each host configuration that accesses the storage system. Each initiator represents the initiator on the host, which will connect to the storage system. There are two types of initiators, Fibre Channel (FC) and iSCSI.

A FC initiator contains the WWN of an HBA on the host. This WWN is not the WWN of the host.

An iSCSI initiator contains the IQN (iSCSI Qualified Name) used by the host, and optionally the CHAP authentication password associated with the host. explains how to configure reverse (two-way) CHAP authentication on the system.

Each initiator is identified by an ID.

The following table lists the attributes for initiators.

Table 7 Initiator attributes

Attribute	Description
`ID`	Host initiator ID.
`Host`	Name of the parent host.
`UID`	FC WWN or iSCSI IQN of the initiator.
`Initiator type`	The type of initiator. Value is one of the following: • `FC` • `iSCSI`
`Ports logged in`	Comma-separated list of array target ports that the initiator is logged into.
`Ignored`	Indicates whether the initiator is ignored for data access to the host. Value is one of the following: • `Yes` — The initiator is ignored. • `No` — The initiator is not ignored.
`Health state`	Health state of the system. The health state code appears in parentheses. Value is one of the following: • `Unknown (0)` — Status is unknown. • `OK (5)` — Working correctly. • `OK BUT (7)` — Working correctly, but there could be a problem. • `Degraded/Warning (10)` — Working and performing all functions, but the performance may not be optimum. • `Minor failure (15)` — Working and performing all functions but overall performance is degraded. This condition has a minor impact on the system and should be remedied at some point, but does not have to be fixed immediately. • `Major failure (20)` — Failing and some or all functions may be degraded or not working. This condition has a significant impact on the system and should be remedied immediately. • `Critical failure (25)` — Failed and recovery may not be possible. This condition has resulted in data loss and should be remedied immediately. • `Non-recoverable error (30)` — Completely failed and cannot be recovered.
`Health details`	Additional health information. See Appendix A, Reference, for health information details.
`CHAP users`	List of CHAP accounts configured for the initiator.
`Source type`	The source initiator type. Values are: • `HPAutotrespass` - HP with Auto-trespass • `OpenNative` (default) - Open native (such as CLARiiON Open) • `SGI` - Silicon Graphics

Table 7 Initiator attributes (continued)

Attribute	Description
	• `HPNoAutotrespass`- HP without Auto-trespass • `Dell` • `FujitsuSiemens` • `Tru64`- Compaq Tru64
`Failover mode`	The failover mode for the initiator. Values are: • `AutoTrespass`- Any media access to the non owning SP is rejected. • `PassiveNotReady`- A command failure during I/O is sent to the non-owning SP. • `DMP`- Quiet trespass on I/O to non owning SP. • `PassiveAlwaysReady`- Some commands, e.g. Test Unit Ready, returns PAR status. • `ALUA`(default) - Initiators are permitted to send I/O to a LUN regardless of which SP actually owns the LUN.
`LUNZ enabled`	Specifies whether LUNZ is enabled. Values are: • `yes` • `no`
`Unit serial number`	Indicates the unity serial number. Values are: • `Array` (default) • `LUN` For SCSI-3 interfaces, the Unity Serial Number page (Vital Product Data page 0x80) reports the serial number for the array or LUN.

Create initiators

Create an FC or iSCSI initiator and assign it to a host configuration.

Format

```
/remote/initiator create –host <value> -uid <value> -type
{iscsi|fc} [-sourceType {HPAutotrespass | OpenNative | SGI |
HPNoAutotrespass | Dell | FujitsuSiemens | Tru64}] [-
failoverMode {AutoTrespass | PassiveNotReady | DMP |
PassiveAlwaysReady | ALUA}] [-lunzEnabled {yes | no}] [-
unitSerialNumber {Array | LUN}]
```

Object qualifier

Qualifier	Description
`-host`	Identifies the host configuration to which to assign the initiator. View host configurations on page 41 explains how to view the IDs of host configurations on the system.

Qualifier	Description
`-uid`	Specifies the FC WWN or the iSCSI IQN of the host to which to assign the initiator.
`-type`	Specifies the type of initiator. Value is one of the following: • `iscsi` • `fc`
`-sourceType`	Specify the source type for the initiator. Valid values are: • `HPAutotrespass` - HP with Auto-trespass • `OpenNative` (default) - Open native (such as CLARiiON Open) • `SGI` - Silicon Graphics • `HPNoAutotrespass`- HP without Auto-trespass • `Dell` • `FujitsuSiemens` • `Tru64`- Compaq Tru64
`-failoverMode`	Specify the failover mode for the initiator. Valid values are: • `AutoTrespass`- Any media access to the non owning SP is rejected. • `PassiveNotReady`- A command failure during I/O is sent to the non-owning SP. • `DMP`- Quiet trespass on I/O to non owning SP. • `PassiveAlwaysReady`- Some commands, e.g. Test Unit Ready, returns PAR status. • `ALUA` (default) - Initiators are permitted to send I/O to a LUN regardless of which SP actually owns the LUN.
`-lunzEnabled`	Set whether LUNZ will be enabled. Valid values are: • `yes` (default) • `no`
`-unitSerialNumber`	Specify the Unit Serial Number. Valid values are: • `Array` (default) • `LUN` For SCSI-3 interfaces, the Unity Serial Number page (Vital Product Data page 0x80) reports the serial number for the array or LUN.

Example 1
The following command creates an FC initiator for host configuration 1014. The FC initiator receives ID 1021:

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /remote/initiator
create -host 1014 -uid "20:00:00:00:C9:29:0F:FD:
10:00:00:00:C9:29:0F:FD" -type fc
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

ID = 1021
Operation completed successfully.
```

Example 2

The following command creates an iSCSI initiator for host configuration Host_3. The iSCSI initiator receives ID 1022:

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! -sslPolicy accept /
remote/initiator create -host Host_3 iqn.1000-05.com.fancy:win-123456
-type iscsi
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

ID = 1022
Operation completed successfully.
```

Example 3

The following command creates an iSCSI initiator for "Host_3" with:

- A source type of "OpenNative"
- A failover mode of "PassiveAlwaysReady"
- LUNZ disabled
- And an "Array" Unit Serial Number

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /remote/initiator
create -host Host_3 -uid iqn.1993-08.com.microsoft:win -type iscsi -
sourceType OpenNative -failoverMode PassiveAlwaysReady -lunzEnabled no
-unitSerialNumber Array
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

ID = HostInitiator_8
Operation completed successfully.
```

View initiators

View a list of initiators. You can filter on the initiator ID, host ID, or whether the initiator is registered.

Format

```
/remote/initiator [{-id <value> | -host <value> | -
unregistered}] show
```

Object qualifier

Qualifier	Description
-id	Identifies the initiator.
-host	Type the ID of a host configuration to view the initiators assigned to the host configuration.
-unregistered	Specifies unregistered initiators.

Example
The following command lists the details of all initiators on the system:

uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /remote/initiator show

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

1: ID                  = HostInitiator_7
   Host                = Host_4
   UID                 = iqn.
1991-05.com.microsoft:cnenfanw4l1c.corp.emc.com
   Initiator type      = iscsi
   Ports logged in     = spb_eth2,spa_eth2
   Ignored             = no
   Health State        = OK (5)
   Health Details      = "The component is operating normally. No
action is required."
   CHAP users          =
   Source type         = Open_Native
   Failover mode       = ALUA
   LUNZ                = yes
   Unit serial number  = Array
```

Change initiator settings

Modify an already created initiator.

Format
/remote/initiator -id <*value*> set [-ignored {yes | no}] [-host <*value*>] [-sourceType {HPAutotrespass | OpenNative | SGI | HPNoAutotrespass | Dell | FujitsuSiemens | Tru64}] [-failoverMode {AutoTrespass | PassiveNotReady | DMP | PassiveAlwaysReady | ALUA}] [-lunzEnabled {yes | no}] [-unitSerialNumber {Array | LUN}]

Object qualifier

Qualifier	Description
-id	Specifies the ID of the initiator

Action qualifier

Qualifier	Description
`-ignored`	Specifies whether the initiator is ignored for data access to the host. Valid values are: • `yes` — The initiator is ignored. • `no` — The initiator is not ignored.
`-host`	Identifies the host configuration to which the initiator is assigned. View host configurations on page 41 explains how to view the IDs of host configurations on the system.
`-sourceType`	Specify the source type for the initiator. Valid values are: • `HPAutotrespass` — HP with Auto-trespass • `OpenNative` — Open native (such as CLARiiON Open) • `SGI` — Silicon Graphics • `HPNoAutotrespass` — HP without Auto-trespass • `Dell` • `FujitsuSiemens` • `Tru64` — Compaq Tru64
`-failoverMode`	Specify the failover mode for the initiator. Valid values are: • `AutoTrespass` — Any media access to the non owning SP is rejected. • `PassiveNotReady` — A command failure during I/O is sent to the non-owning SP. • `DMP` — Quiet trespass on I/O to non owning SP. • `PassiveAlwaysReady` — Some commands, e.g. Test Unit Ready, returns PAR status. • `ALUA` — Initiators are permitted to send I/O to a LUN regardless of which SP actually owns the LUN.
`-lunzEnabled`	Set whether LUNZ will be enabled. Valid values are: • `yes` • `no`
`-unitSerialNumber`	Specify the Unit Serial Number. Valid values are: • `Array` • `LUN` For SCSI-3 interfaces, the Unity Serial Number page (Vital Product Data page 0x80) reports the serial number for the array or LUN.
`-force`	Specify to bypass the validation of setting a new host when there are already storage resources associated with the host and attached to the initiator.

Example

The following command changes the source type, failover mode, LUNZ settings, and Unit Serial Number of the initiator:

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /remote/initiator -
id HostInitiator_6 set -sourceType HPAutotrespass -failoverMode
PassiveNotReady -lunzEnabled yes -unitSerialNumber Array
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

Operation completed successfully.
```

Manage VMware vCenter

Manage VMware vCenter servers.

The following table lists the attributes for VMware vCenter.

Table 8 VMware vCenter attributes

Attribute	Description
ID	ID of the VMware virtual center
Address	Domain name or IP address of VMware vCenter.
User name	Name of the user account on the VMware vCenter.
Password	Password of the user account on the VMware vCenter.
Description	Description of the VMware vCenter.
VASA provider state	Indicates whether the system is registered as a VASA provider in vCenter. Values are: • Registered • Not registered • Not supported **Note** Automatic VASA registration is not supported on vSphere versions earlier than 6.0. The storage system can be registered as a VASA provider with only one vCenter at a time.
Local username	The username of the local account that vSphere will use to register the system as a VASA provider.

Table 8 VMware vCenter attributes (continued)

Attribute	Description
	Note It is recommended that you create a new user with the `/user/account` command and set the role to *vmadmin*.
`Local password`	The password of the local account that vSphere will use to register the system as a VASA provider.

Create VMware vCenter

Adds the vCenter credentials and discovers any ESXi host managed by that vCenter. The vCenter credentials are stored in the storage system. In order to execute this command, the user must have account on the storage system.

Format

```
/virt/vmw/vc create -addr <value> -username <value> {-passwd
<value> | -passwdSecure} [-descr <value>] [-
registerVasaProvider {yes -localUsername <value> {-localPasswd
<value> | -localPasswdSecure} | no}]
```

Action qualifier

Qualifier	Description
`-addr`	Domain name or IP address or domain name of the VMware vCenter.
`-username`	Specify the username used to access the VMware vCenter.
`-passwd`	Specify the password used to access the VMware vCenter.
`-passwdSecure`	Specify the password in secure mode. The user will be prompted to input the password.
`-descr`	Specify the description of the VMware vCenter server.
`-registerVasaProvider`	Specify to register the system as a VASA provider with this vCenter server. Valid values are: • `yes` • `no`
`-localUsername`	Specify the username of the system account that will be used by vCenter to register the system as a VASA provider.

Qualifier	Description
	Note It is recommended that you create a new user with the `/user/account` command and set the role to *vmadmin*. The storage system can be registered as a VASA provider with only one vCenter at a time.
`-localPasswd`	Specify the password of the system account that will be used by vCenter to register the system as a VASA provider.
`-localPasswdSecure`	Specify the VASA password in secure mode, which requires the user to input the password when prompted.

Example 1

The following command adds virtual center credentials:

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /virt/vmw/vc create
-addr 10.11.11.111 -username administrator@vsphere.local -passwd xxx -
descr "Add vCenter"
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

ID = VC_1
Operation completed successfully
```

Example 2

The following command adds a vCenter and registers the storage system as a VASA provider.

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /virt/vmw/vc create
-address 10.11.11.111 -username root -passwd xxx -descr "Add virtual
center" -registerVasaProvider yes -localUsername admin -localPasswd
Password321
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

ID = VC_1
Operation completed successfully
```

Set the credentials or description of an existing vCenter server

Modifies the credentials or description of the existing vCenter server. In order to execute this command the user must have an account on the storage system.

Format

```
/virt/vmw/vc -id <value> set [-addr <value>] [-username <value>
{-passwd <value> | -passwdSecure} ] [-descr <value>]
```

Object qualifier

Qualifier	Description
-id	Identifies the VMware vCenter server.

Action qualifier

Qualifier	Description
-addr	Specifies the new IP address or domain name of the VMware vCenter server.
-username	Specifies the username.
-passwd	Specifies the password.
-passwdSecure	Specifies the password in secure mode - the user will be prompted to input the password.
-descr	Specifies the new description of the VMware vCenter server.

Example

The following command specifies the new description of the VMware vCenter server:

```
uemcli /virt/vmw/vc -id VC_1 set -descr "This vCenter manages 2 ESXi hosts"
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

ID = VC_1
Operation completed successfully.
```

Delete an existing vCenter server

Removes an existing VMware vCenter server and its associated ESXi hosts.

Note

If the Unity system is registered as a VASA provider in vCenter and you delete the vCenter from Unity, the Unity system will be unregistered as a VASA provider from vCenter.

Format

```
/virt/vmw/vc -id <value> delete
```

Object qualifier

Qualifier	Description
-id	Identifies the VMware vCenter server.

Example

The following example deletes an existing vCenter server and any of its associated ESXi hosts.

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /virt/vmw/vc -id
VC_1 delete
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

Operation completed successfully
```

View all vCenter servers

Displays a list of configured VMware vCenter servers.

Format

`/virt/vmw/vc [-id <value>] show`

Object qualifier

Qualifier	Description
`-id`	Identifies the VMware vCenter server.

Example

The following example shows a list of all vCenter servers.

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /virt/vmw/vc show
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

1:     ID                       = VC_1
       Address                  = 10.1.1.1
       Description              = This vCenter manages 2 ESXi hosts
       VASA provider state      = yes
```

Refresh all vCenter servers

Rescan details of all configured VMware vCenter servers.

Format

`/virt/vmw/vc refresh [-scanHardware]`

Object qualifier

Qualifier	Description
`-id`	Specify the ID of the vCenter. If not specified, all attached vCenters are refreshed.
`-scanHardware`	Specify to rescan hardware changes (this takes additional time).

Example

The following example rescans all vCenters.

uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /virt/vmw/vc refresh -scanHardware

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

Operation completed successfully.
```

Manage ESXi hosts

Manage VMware ESXi hosts.

The following table lists the attributes for ESXi hosts.

Table 9 ESXi host attributes

Attribute	Description
ID	ID of the ESXi host.
Name	Name of the ESXi host.
Address	Domain name or IP address of ESXi host.
Virtual center	Identifier of the VMware VCenter server managing the ESXi host.
Username	Name of the user account on the ESXi host.
Password	Password of the user account on the ESXi host.
Description	Description of the ESXi host.
NFSv4 supported	Indicates if the NFSv4 protocol is supported for the host. Valid values are: • yes • no
NFS username	Displays the NFS user authentication information configured for the ESXi host. The same username should be configured on the VMware NFS datastore in order to enable secure NFS access with Kerberos for that datastore.

Create an ESXi host

Adds a VMware ESXi host.

Format

```
/virt/vmw/esx create -addr <value> { -vc <value> | -username
<value> {-passwd <value> | -passwdSecure} } [ -descr
<value> ] ] [ -resolveConflicts { yes | no } ]
```

Action qualifier

Qualifier	Description
-addr	Domain name or IP address of the ESXi host.
-vc	Identifies the VMware vCenter server.
-username	Specifies the username used to access the VMware ESXi host.
-passwd	Specifies the password used to access the VMware ESXi host.
-passwdSecure	Specifies the password in secure mode - the user will be prompted to input the password.
-descr	Specifies the description of the VMware ESXi host.
-resolveConflicts	Specifies the option to resolve IP address or initiator conflicts interactively. Valid values are yes or no (default).

Example 1

uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /virt/vmw/esx create -addr 10.1.1.1 -username root -passwd xxx -descr "My ESXi host"

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

ID = ESX_1
Operation completed successfully
```

Example 2

uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /virt/vmw/esx create -addr 10.1.1.1 -vc VMwareVC_12 -resolveConflicts yes

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

The ESX host to be created has IP addresses and/or Initiators
already present in an existing host.
The ID of the existing host is: Host_12
The IP addresses in conflict are: 10.14.12.219, 10.14.12.220
The Initiators in conflicts are: iqn.1998-01.com.vmware:test1-1,
iqn.1998-01.com.vmware:test1-2

WARNING, the existing host has IP addresses and/or Initiators not
found in the ESX host to be created. If you continue with the ESX
host creation, those IP addresses and/or Initiators will be removed
and can no longer be used for storage access.
The IP address not in the ESX host are: 10.14.12.217, 10.14.12.218
The Initiators not in the ESX host are: iqn.
1998-01.com.vmware:test1-3

Do you want to convert the existing host to the ESX host?
Yes / no:yes

ID = ESX_1
Operation completed successfully
```

Change ESXi host credentials

Changes ESXi host credentials and/or description. In order to execute this command the user must have account on the storage system.

Format

```
/virt/vmw/esx -id <value> set [ -descr <value> ] [ -username
<value> { -passwd <value> | -passwdSecure } ] [ -addr <value> ]
```

Object qualifier

Qualifier	Description
-id	Identifies the VMware ESXi host.

Action qualifier

Qualifier	Description
-descr	Specifies the comment or description.
-username	Specifies the username used to access the VMware ESXi host.
-passwd	Specifies the password used to access the VMware ESXi host.
-passwdSecure	Specifies the new password in secure mode - the user will be prompted to input the password.
-addr	Specifies the domain name or IP address of the ESXi host in order for Unisphere to contact the ESXi host directly. **Note** This is only applicable for standalone ESXi hosts.

Example

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /virt/vmw/esx -id
ESX_1 set -descr "Changing ESXi host description"
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

ID = ESX_1
Operation completed successfully.
```

Delete ESXi host credentials

Deletes ESXi host credentials. This will also remove access from the specified host to any VMware datastores or protocol endpoints that are associated with it.

Format

```
/virt/vmw/esx -id <value> delete
```

Object qualifier

Qualifier	Description
-id	Identifies the ESXi host.

Example

uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /virt/vmw/esx -id ESX_1 delete

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

Operation completed successfully.
```

View all existing ESXi hosts

Displays a list of all configured VMware ESXi hosts.

Format

/virt/vmw/esx [{-id *<value>* | -vc *<value>*}] show

Object qualifier

Qualifier	Description
-id	Identifies the VMware ESXi host.
-vc	Identifies the VMware vCenter server.

Example

The following example shows how to display all of the ESXi hosts on the vCenter connected to the system.

uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /virt/vmw/esx -vc VC_1 show

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

1:     ID              = ESX_1
       Name            = nlpc12240.aa.bb.com
       vCenter         = VC_1
       Address         = 10.10.10.100
       Description     =
       NFSv4 supported = yes
       NFS username    = root


2:     ID              = ESX_2
       Name            = nlpc12241.xx.yy.com
       vCenter         = VC_1
       Address         = 10.10.10.101
       NFSv4 supported = no
       NFS username    =
```

Discover all ESXi hosts

Lists all VMware ESXi hosts on the specified VMware vCenter server.

Format

```
/virt/vmw/esx discover { -vc <value> | -vcAddr <value> -
username <value> {-passwd <value> | -passwdSecure} } [ -
createAll ]
```

Action qualifier

Qualifier	Description
-vc	Identifies the existing VMware vCenter.
-vcAddr	IP address or domain name of the VMware vCenter.
-username	Specifies the name of the VMware vCenter.
-passwd	Specifies the password of the VMware vCenter
-passwdSecure	Specifies the password in secure mode - the user will be prompted to input the password.
-createAll	Adds all discovered ESXi hosts automatically.

Example

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /virt/vmw/esx
discover -vc VC_1
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

1:     Name    = nlpc12240.us.dg.com

2:     Name    = nlpc12241.us.dg.com

Operation completed successfully
```

Refresh an ESXi host

Rescans details of a VMware ESXi host.

Format

```
/virt/vmw/esx [-id <value>] refresh [-scanHardware]
```

Object qualifier

Qualifier	Description
-id	Identifies the ESXi host. If an ID is not specified, all virtualization objects are rescanned.

Action qualifier

Qualifier	Description
`-scanHardware`	Specify to rescan hardware changes also (takes additional time).

Example

The following command rescans the hardware to discover additional ESXi hosts.

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /virt/vmw/esx
refresh -scanHardware
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection


Operation completed successfully.
```

Manage capability profiles

A capability profile is a group of storage capabilities that are applicable for VVol datastores. You must create one or more capability profiles before creating VVol datastores.

Capabilities are automatically derived from the underlying storage pool and are determined by the pool properties. Usage tags are assigned by the storage admin.

There are three ways to profile storage capabilities for a pool:

Table 10 Storage capabilities

Capability name	Description

Service level-based provisioning (physical deployments)	Expected service level for the pool:
	• Platinum
	  ▪ Single-tiered Flash pool
	• Gold
	  ▪ Multitiered pool with a mix of Flash and SAS drives
	  ▪ Single-tiered pools with SAS RAID 10
	• Silver
	  ▪ Single-tiered pools with SAS RAID 5 or RAID 6
	  ▪ Multitiered pools with a mix of SAS and NL-SAS
	• Bronze
	  ▪ Single-tiered pools with NL-SAS
	  ▪ Multitiered pools with a mix of Flash and NL-SAS

Service level-based provisioning (virtual deployments)	Expected service level for a virtual pool: • Gold

Table 10 Storage capabilities (continued)

Capability name	Description
	▪ Multitiered pool with a mix of Extreme Performance and Performance tiers ▪ Single-tiered Extreme Performance pool • Silver ▪ Multitiered pool with a mix of Extreme Performance, Performance, and Capacity tiers ▪ Multitiered pool with a mix of Performance and Capacity tiers ▪ Single-tiered Performance pool • Bronze ▪ Multitiered pool with a mix of Extreme Performance and Capacity tiers ▪ Single-tiered Capacity pool
Usage tags	Usage tags can be applied to capability profiles to designate them and their associated VVol datastores for a particular use. For example, a VVol datastore may be tagged for VVols and VMs that support a particular application. The virtualization administrator and storage administrator should collaborate to define these usage tags.
Storage properties	Supported storage properties include: • Drive type: ▪ Extreme Performance [Flash] ▪ Performance [SAS] ▪ Capacity [NL-SAS] ▪ Multitier [mixed] ▪ Extreme Multitier [mixed with Flash] • RAID type (physical deployments only): ▪ RAID5 ▪ RAID6 ▪ RAID10 ▪ Mixed • FAST Cache (physical deployments only): ▪ Enabled ▪ Disabled • FAST VP tiering policy: ▪ Highest Available Tier ▪ Start High then Auto-Tier

Table 10 Storage capabilities (continued)

Capability name	Description
	▪ Auto-Tier ▪ Lowest Available Tier • Space Efficiency: ▪ Thick ▪ Thin

Table 11 Capability profile attributes

Attribute	Description
`ID`	Capability profile identifier.
`Name`	Capability profile name.
`Description`	Capability profile description.
`VMware UUID`	VMware UUID of the capability profile.
`Storage pool`	Associated storage pool identifier.
`Service level`	Service level of the underlying storage pool. Valid values are: • `Platinum` • `Gold` • `Silver` • `Bronze`
`Usage tag`	Comma-separated list of user-defined tags. Each tag is an alphanumeric string value.
`Drive type`	Specifies the drive type of the underlying storage pool. Valid values are: • `CapacityTier` • `PerformanceTier` • `ExtremePerformanceTier` • `MultiTier` • `ExtremeMultiTier`
`RAID level` (physical deployments only)	Specifies the RAID level of the underlying storage pool. Valid values are: • `RAID5` • `RAID10` • `RAID6` • `Mixed`

Table 11 Capability profile attributes (continued)

Attribute	Description
`FAST Cache` (physical deployments only)	Indicates whether or not FAST Cache is enabled on the underlying storage pool. Valid values are: • `On` • `Off`
`FAST VP policy`	Comma-separated list of FAST VP storage policies for the underlying storage pool. Valid values are: • `Start high then auto-tier` • `Auto-tier` • `Highest available tier` • `Lowest available tier`
`Space efficiency`	Comma-separated list of available space efficiency policies for the underlying storage pool. Valid values are: • `Thick` • `Thin`
`Health state`	Health state.
`Health details`	Additional health information.

Create a capability profile

Create a capability profile for VVol datastores.

Format

```
/stor/config/cp create [-async] -name <value> [-descr <value>]
-pool <value> [-usageTag <value>]
```

Action qualifier

Qualifier	Description
`-async`	Run the operation in asynchronous mode.
`-name`	Type a name for the capability profile. **Note** The name may contain alphanumeric values, a hyphen, an underscore, and a period. It cannot start with hyphen or period, and cannot consist only of digits.
`-descr`	Type a description for the capability profile.
`-pool`	Specify the identifier of the storage pool the capability profile is based on.

Qualifier	Description
-usageTag	Type a comma-separated list of user-specified usage tags. Each tag is an alphanumeric string value.

Example

The following command creates a capability profile with these settings:

- Specifies a capability profile name of "CapabilityProfile1"
- Specifies that the capability profile is based on "pool_1"
- Specifies the usage tag as "Production"
- Not specified to be created in asynchronous mode

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /stor/config/cp
create -name "CapabilityProfile1" -pool pool_1 -usageTag "Production"
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

ID = cp_1
Operation completed successfully.
```

View capability profiles

Displays a list of existing capability profiles and their characteristics.

Format

```
/stor/config/cp [-id <value>] show
```

Object qualifier

Qualifier	Description
-id	Type the ID of the capability profile.

Example

The following command displays a list of existing capability profiles and their characteristics.

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /stor/config/cp show
-detail
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

       ID               = cp_1
       Name             = CapabilityProfile1
       Description      =
       VMware UUID      = 550e8400-e29b-41d4-a716-446655440000
       Storage pool     = pool_1
       Service level    = Gold
       Usage tag        = Exchange, OLTP
       Drive type       = ExtremeMultiTier
       RAID level       = Mixed
       FAST Cache       = Off
```

```
       FAST VP policy   = Start high then auto-tier, Auto-tier,
Highest available tier,
                          Lowest available tier
       Space efficiency = Thin, Thick
       Health state     = OK (5)
       Health details   = "The component is operating
                           normally. No action is required."
```

Change capability profiles

Modify an existing capability profile.

Format

```
/stor/config/cp -id <value> set [-async] [-name <value>] [-
descr <value>] [{-addUsageTag <value> | -removeUsageTag
<value>}]
```

Object qualifier

Qualifier	Description
-id	Type the ID of the capability profile to be modified.

Action qualifier

Qualifier	Description
-async	Run the operation in asynchronous mode.
-name	Type a name for the capability profile. **Note** The name may contain alphanumeric values, a hyphen, an underscore, and a period. It cannot start with hyphen or period, and cannot consist only of digits.
-descr	Type a description for the capability profile.
-addUsageTag	Comma-separated list of user-specified usage tags to be added to the specified capability profile. Each tag is an alphanumeric string value.
-removeUsageTag	Comma-separated list of user-specified usage tags to be removed from the specified capability profile. Each tag is an alphanumeric string value.

Example

The following command changes the name of capability profile "cp_1".

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /stor/config/cp -id
cp_1 set -name "CapabilityProfile2"
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection
```

```
ID = cp_1
Operation completed successfully.
```

Delete capability profiles

Deletes specified capability profiles.

Format

```
/stor/config/cp [-id <value>] delete [-async]
```

Object qualifier

Qualifier	Description
-id	Type the ID of the capability profile.

Action qualifier

Qualifier	Description
-async	Run the operation in asynchronous mode.

Example

The following command deletes capability profile cp_1.

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /stor/config/cp -id
cp_1 delete
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

Operation completed successfully.
```

Manage VMware protocol endpoints

Protocol Endpoints (PEs) are access points for ESX/ESXi host communication to the storage system. These endpoints establish a datapath on-demand for virtual machines and their respective VVol datastores. I/O from VMs is communicated through the PE to the VVol datastore on the storage system. A single protocol endpoint can multiplex I/O requests from a large number of VM clients to their virtual volumes.

NAS protocol endpoints are created and managed on the storage system and correspond to a specific NFS-based NAS server. It is recommended that you enable at least two NAS servers for VVols, one for each SP, for high availability. A File VVol will be bound to the associated NAS PE every time that VM is powered on. When the VM is powered off, VVols is unbound from the PE.

SCSI protocol endpoints correspond to a specific iSCSI interface or Fibre Channel connection. The Block VVol will be bound to the associated SCSI PE every time that the VM is powered on. When the VM is powered off, the PE is unbound. SCSI protocol endpoints are like LUN mount points that allow I/O access to VVols from the ESXi host to the storage system.

Table 12 Protocol endpoint attributes

Attribute	Description
`ID`	VMware protocol endpoint identifier.
`Name`	Protocol endpoint name.
`Type`	Type of protocol endpoint. Valid values are: • `SCSI` • `NAS`
`VMware UUID`	VMware UUID of the protocol endpoint.
`Export path` (NAS PEs only)	Export path to the PE.
`IP address`	IP address of the NAS server for File PEs.
`WWN`	The World Wide Name for Block PEs.
`Default SP`	Identifier for the preferred SP. Valid values are: • `SPA` • `SPB`
`Current SP`	Identifier for the current SP. Valid values are: • `SPA` • `SPB`
`NAS server`	Identifier of the associated NAS server for NAS PEs.
`VMware NAS PE server` (NAS PEs only)	ID of the corresponding VMware NAS PE server.
`VVol datastore` (NAS PEs only)	ID of the VVol datastore using the PE.
`Host` (SCSI PEs only)	Comma-separated list of identifiers for hosts that use the PE.
`LUN ID`	Logical Unit Number for the protocol endpoint on the host.
`Health state`	Health state.
`Health details`	Additional health information.

View protocol endpoints

Displays a list of existing protocol endpoints and their characteristics.

Format

```
/stor/prov/vmware/pe [-id <value>] show
```

Object qualifier

Qualifier	Description
`-id`	Type the ID of the protocol endpoint.

Example

The following example shows the detail for all protocol endpoints on the system.

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /stor/prov/vmware/pe
show -detail
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

1:     ID                   = rfc4122.60060160-
ca30-3c00-962b-87806445241a
       Name                 = scsi_pe_1
       Type                 = SCSI
       VMware UUID          = rfc4122.60060160-
ca30-3c00-962b-87806445241a
       Export path          =
       IP address           =
       WWN                  = 60:06:01:60:CA:30:3C:00:96:2B:
87:80:64:45:24:1A
       Default SP           = SPA
       Current SP           = SPA
       NAS Server           =
       VMware NAS PE server =
       VVol datastore       =
       Host                 = Host_1
       LUN ID               =
       Health state         = OK (5)
       Health details       = "The protocol endpoint is operating
normally. No action is required."
```

Manage VVol datastores

VVols reside in VVol datastores, also known as storage containers, which are comprised of storage allocations from one or more capability profiles. Capability profiles are built on top of one or more underlying storage pools. You can create VVol datastores based on one or more capability profiles and then allocate a specific amount of space from the capability profile to the VVol datastore.

Each VVol datastore has one or more capability profiles that describe its performance and capacity characteristics, such as drive type, FAST VP tiering policy, and space efficiency policy (thick or thin). These characteristics are derived based on the underlying storage pool. When a virtual volume is created in vSphere, it is assigned a storage policy profile. vSphere filters the compatible and incompatible available VVol datastores (from one or more storage systems) when the VVol is being created based on these profiles. Only VVol datastores that support the storage policy profile are considered compatible storage containers for deploying the VVol.

Table 13 VVol datastore attributes

Attribute	Description
`ID`	VVol datastore identifier.
`Name`	VVol datastore name.
`Description`	VVol datastore description.
`VMware UUID`	VWware UUID of the VVol datastore.
`Type`	Type of VVol datastore. Valid values are:

Table 13 VVol datastore attributes (continued)

Attribute	Description
	• File • Block

Health state	Health state of the VVol datastore. Value is one of the following:
	• Unknown (0) - Health is unknown.
	• OK (5) - Operating normally.
	• OK BUT (7)
	  ▪ Storage resource allocation from one or more pools has exceeded the 85% threshold.
	  ▪ Storage resource allocation from one or more pools has exceeded the 95% threshold.
	• Degraded/Warning (10)
	  ▪ Pool performance is degraded on one or more of the underlying storage pools for the virtual volume.
	  ▪ Storage resource allocation from one or more pools has exceeded the 95% threshold, and the storage resource is oversubscribed.
	• Major failure (20)
	  ▪ The storage resource has failed due to one or more failed storage pools.
	  ▪ The storage resource is unavailable due to one or more unavailable servers.
	  ▪ The storage resource is unavailable and requires a Storage Integrity Check.
	• Critical failure (25) - One or more of the underlying storage pools for a virtual volume is offline.
	• Non-recoverable error (30) - Resource unavailable due to one or more unavailable storage pools.

Health details	Detailed health state for the VVol datastore.
Capability profile	Comma-separated list of identifiers of capability profiles supported by the VVol datastore. Each identifier with a " (Not used) " suffix indicates that this profile can be removed from the VVol datastore.
Storage pool ID	Comma-separated list of identifiers of storage pools used for the VVol datastore.
Total capacity	Total capacity of the VVol datastore.
Total current allocation	Total current allocation of the VVol datastore in all associated storage pools.
Total used capacity	Total used capacity of the VVol datastore.
Creation time	Time when the VVol datastore was created.

Table 13 VVol datastore attributes (continued)

Attribute	Description
`Hosts`	Hosts that have access to the datastore.
`Last modified time`	Time when the VVol datastore was last modified.

Create VVol datastores

Create a datastore for VMware VVols.

Format

```
/stor/prov/vmware/vvolds create [-async] -name <value> [-descr <value>] -cp <value> -size <value> -type { block | file } [-hosts <value>]
```

Action qualifier

Qualifier	Description
`-async`	Run the operation in asynchronous mode.
`-name`	Type a name for the VVol datastore. **Note** The name may contain alphanumeric values, a hyphen, an underscore, and a period. It cannot start with hyphen or period, and cannot consist only of digits.
`-descr`	Type a brief description for the VVol datastore.
`-cp`	Specify the list of identifiers of capability profiles supported by the VVol datastore.
`-size`	Specify the list of allocation sizes. Specify one allocation for the amount of total space available for VVol provisioning on the VVol datastore for the specified capability profile. If there are multiple capability profiles, the list should include allocation size respective to each capability profile.
`-type`	Specify the VVol datastore type. Valid values are: • `block` • `file`
`-hosts`	Specify the comma-separated list of hosts that will have access to the VVol datastore. For a list of eligible hosts, refer to View host configurations on page 41.

Example

The following command creates a VVol datastore with these settings:

- A VVol datastore name of "Engineering department"
- Associates the "cp_1" and "cp_2" capability profiles with this VVol datastore
- Allocates 10 GBs and 12 GBs from capability profiles cp_1 and cp_2, respectively, to the VVol datastore

- Grants access for "Host_1" and "Host_2" to the datastore

uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /stor/prov/vmware/vvolds create -name "Engineering department" -cp cp_1,cp_2 -size 10G, 12G -type file -hosts "Host_1,Host_2"

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

ID = res_1
Operation completed successfully.
```

View VVol datastores

Display a list of existing VVol datastores and their characteristics.

Format

```
/stor/prov/vmware/vvolds [-id <value>] show
```

Object qualifiers

Qualifier	Description
-id	Type the ID of the VVol datastore.

Example

The following command displays a list of VVol datastores and their characteristics.

uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /stor/prov/vmware/vvolds show -detail

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

1:     ID                        = res_1
       Name                      = Performance
       Description               =
       VMware UUID               = 550e8400-e29b-41d4-
a716-446655440000
       Type                      = Block
       Health state              = OK (5)
       Health details            = "The component is operating
                                    normally. No action is required."
       Capability profile        = cp_1, cp_2 (Not used)
       Storage pool              = pool_1,pool_3
       Total capacity            = 128849018880 (120G)
       Total current allocation = 12884901888 (12G)
       Total used capacity       = 1073741824 (1G)
       Hosts                     = Host_1
       Creation time             = 2015-12-21 12:55:32
       Last modified time        = 2016-01-15 10:31:56

2:     ID                        = res_2
       Name                      = engineering
       Description               =
       VMware UUID               = rfc4122.534e0655-
f5a3-41d7-8124-9d53be5d0c0d
       Type                      = file
       Health state              = OK (5)
       Health details            = "The component is operating
                                    normally. No action is
```

```
required."
       Capability profile       = cp_1, cp_2
       Storage pool             = pool_1, pool_2
       Total capacity           = 644245094400 (600.0G)
       Total current allocation = 0
       Total used capacity      = 0
       Creation time            = 2015-06-20 01:48:54
       Last modified time       = 2015-06-20 01:48:54
```

Manage VVol datastore allocation

Manage the allocation of storage to VVol datastores.

Table 14 VVol datastore allocation attributes

Attribute	Description
ID	VVol datastore allocation identifier.
VVol datastore	VVol datastore identifier.
Capability profile	Identifier of the associated capability profile.
Storage pool	Comma-separated list of identifiers of storage pools associated with the capability profile.
Size	Amount of total space available for VVol provisioning for a particular capability profile on the VVol datastore.
Current allocation	Quantity of primary storage currently allocated for the VVol datastore for VVols provisioned with a particular capability profile on the VVol datastore.
Size used	Amount of space used by virtual volumes provisioned with a particular capability profile on the VVol datastore.
Health state	Health state of the VVol datastore allocation.
Health details	Additional health information.

View VVol datastore allocation details

Displays existing VVol datastore allocations.

Format

```
/stor/prov/vmware/vvolds/alloc {-id <value> | -vvolds <value>
[{-pool <value> | -cp <value>}]} show
```

Object qualifier

Qualifier	Description
-id	Type the allocation identifier of the VVol datastore.
-vvolds	Type the ID of the VVol datastore.
-pool	Type the ID of the storage pool.

Qualifier	Description
-cp	Type the ID of the capability profile.

Note

To obtain the ID of the VVol datastore and it's associated pool and capability profile IDs, refer to View VVol datastores on page 72.

Example

The following command shows the allocation details for the VVol datastore "vvol_1" from pool "pool_1", including associated capability profile IDs, current size of the storage pool, and current size allocated to the VVol datastore from the storage pool.

```
uemcli /stor/prov/vmware/vvolds/alloc -vvolds vvolds_1 -pool pool_1
show -detail
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

1:     ID                 = cpa_1
       VVol datastore     = res_1
       Capability profile = cp_1
       Storage pool       = pool_1
       Size               = 128849018880 (120G)
       Current allocation = 12884901888 (12G)
       Size used          = 1073741824 (1G)
       Health state       = OK (5)
       Health details     = "The component is operating
                             normally. No action is required."
```

Change VVol datastores

Modify an existing VVol datastore.

Format

```
/stor/prov/vmware/vvolds -id <value> set [-async] [-name
<value>] [-descr <value>] [{-addCp <value> -size <value> | -
modifyCp <value> -size <value> | -removeCp <value>}] [-hosts
<value> [-force]]
```

Object qualifier

Qualifier	Description
-id	Type the ID of the VVol datastore to be modified.

Action qualifier

Qualifier	Description
-async	Run the operation in asynchronous mode.
-name	Type a name for the VVol datastore.

Qualifier	Description
	Note The name may contain alphanumeric values, a hyphen, an underscore, and a period. It cannot start with hyphen or period, and cannot consist only of digits.
-descr	Type a new description for the VVol datastore.
-addCp	Type the list of identifiers of new capability profiles the VVol datastore will support.
-modifyCp	Type the list of identifiers of capability profiles already supported by the VVol datastore and specify the new allocated sizes for each.
-size	Specify the list of allocation sizes. Specify one allocation for the amount of total space available for VVol provisioning on the VVol datastore for the specified capability profile. If there are multiple capability profiles, the list should include allocation size respective to each capability profile.
-removeCp	Type the list of identifiers of capability profiles you would like to remove from the VVol datastore. **Note** This command can only used on capability profiles that are not currently in use by existing virtual volumes.
-hosts	Type the list of comma-separated hosts that will have access to the VVol datastore.
-force	Type to unconditionally unbind all virtual volumes that are currently bound to a protocol endpoint associated with a particular host. **Note** If host access is changed or removed for a VVol datastore, the associated protocol endpoints are automatically unbound.

Example

The following command modifies the following settings of a VVol datastore:

- Changes the description of the VVol datastore to "My new description"
- Changes the name of the VVol datastore to "MyNewName"
- Associates the capability profile "cp_1" with VVol datastore "res_1"
- Allocates 10 GBs of space from the pool to capability profile "cp_1"

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /stor/prov/vmware/
vvolds -id res_1 set -name MyNewName -descr "My new description" -
addCp cp_1 -size 10G
```

```
Storage system address: 10.0.0.1
Storage system port: 443
```

```
HTTPS connection

ID = res_1
Operation completed successfully.
```

Delete VVol datastores

Deletes specified VVol datastores and their associated virtual volumes.

Format

```
/stor/prov/vmware/vvolds [-id <value>] delete [-async] [-force
{ yes | no}]
```

Object qualifier

Qualifier	Description
-id	Type the ID of the VVol datastore.

Action qualifier

Qualifier	Description
-force	Delete the VVol datastore and any of its associated VVols. Valid values are: • yes • no
-async	Run the operation in asynchronous mode.

Example

The following command deletes VVol datastore res_1 as well as its virtual volumes.

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /stor/prov/vmware/
vvolds -id res_1 delete -force yes
```

```
Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

Operation completed successfully.
```

Manage VVol objects

Virtual volumes are encapsulations of virtual machine files, virtual disks, and their derivatives. There are several types of VVol objects that correspond to an individual virtual volume, including a VMDK VVol (data VVol), Config VVol, Memory VVol, and Swap VVol.

Table 15 Types of VVols

VMDK (Data) VVol	The VMDK VVol, displayed as **Data** VVol in Unisphere, contains the vDisk file, or the hard disk drive, for the VM.

Table 15 Types of VVols (continued)

Config VVol	The **Config** VVol contains settings, configuration, and state information for the VM. This includes .vmx, nvram, and log files.
Memory VVol	The **Memory** VVol contains a complete copy of the VM memory as part of a with-memory VM snapshot.
Swap VVol	The **Swap** VVol is created when VMs are powered on and contain copies of the VM memory pages that are not retained in memory.

Table 16 VVol attributes

Attribute	Description
`ID`	Virtual volume identifier.
`Name`	Virtual volume name.
`Type`	Type of virtual volume. Valid values are: • `Data` • `Config` • `Memory` • `Swap` • `Other`
`Replica type`	Virtual volume replica type. Valid values are: • `Base` • `Prepared Snap` • `Ready Snap` • `Fast-Clone`
`Parent`	Identifier of the base/parent virtual volume for the snap, prepared snap, or fast-clone.
`Health state`	Health state of the virtual volume.
`Health details`	Additional health information for the virtual volume.
`Datastore`	Identifier of the datastore associated with the virtual volume.
`Storage pool`	Identifier of the storage pool that contains the virtual volume.
`Capability profile`	Identifier of the capability profile associated with the virtual volume.
`Policy profile`	Name of the VMware vSphere policy profile.
`Compliant`	Indicates whether the virtual volume is compliant with the VMware vSphere policy profile.

Table 16 VVol attributes (continued)

Attribute	Description
`Size`	Size of the virtual volume.
`Current allocation`	Total current allocation of the virtual volume.
`Bound to`	Comma-separated list of protocol endpoint identifiers to which the virtual volume is bound. An empty value indicates an unbound virtual volume.
`Binding details`	Binding details of the protocol endpoint to which the virtual volume is bound. • For virtual volumes bound to NFS protocol endpoints, this displays the full NFS paths. • For virtual volumes bound to iSCSI protocol endpoints, this displays the virtual volume iSCSI secondary ID. • For unbound virtual volumes, this value is empty.
`Virtual machine`	Identifier of the virtual machine.
`VM hard disk`	Name of the associated VM hard disk.

View VVol objects

Display a list of existing VVol datastores and their characteristics.

Format

```
/stor/prov/vmware/vvol {[-id <value> | [-vm <value>] [-cp
<value>] [-pool <value>] [-datastore <value>] [-pe <value>] [-
parent <value>] [-bound] [-noncompliant] } show
```

Object qualifier

Qualifier	Description
`-id`	Type the ID of the virtual volume.
`-vm`	Type the ID of the associated VM for the virtual volume.
`-cp`	Type the ID of the capability profile associated with the virtual volume.
`-pool`	Type the ID of the storage pool that contains the virtual volume.
`-datastore`	Type the ID of the associated VVol datastore.
`-pe`	Type the ID of the protocol endpoint for which you want to see bound virtual volumes.
`-parent`	Type the ID of the parent virtual volume.
`-bound`	Specify in order to display a list of only bound virtual volumes.
`-noncompliant`	Specify in order to display only a list of virtual volumes not compliant with their respective VMware policy profiles.

Example

The following example displays the details of all VVols for the VM with the ID VM_1.

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /stor/prov/vmware/
vvol -vm VM_1 show -detail
```

```
Storage system address: 10.64.75.201
Storage system port: 443
HTTPS connection

1:     ID                 = rfc4122.de305d54-75b4-431b-adb2-
eb6b9e546014
       Name               = Hard disk 1
       Type               = Data
       Replica type       = Base
       Parent             =
       Health state       = OK (5)
       Health details     = "The component is operating normally.
No action is required."
       Datastore          = res_1
       Storage pool       = pool_1
       Capability profile = cp_1
       Policy profile     = VMware policy profile
       Compliant          = yes
       Size               = 1073741824 (1G)
       Thin               = yes
       Current allocation = 107374182 (100M)
       Bound to           = NASPE_1
       Binding details    = 192.168.3.3:/vvol1
       Virtual machine    = VM_1
       VM hard disk       = VM Hard Disk 1
```

Delete VVol objects

Deletes the specified existing VVol objects.

Note

Deletion of VVol objects must be exclusively confirmed by the user. The following confirmation message will display:

```
Virtual volume deletion will also unbind and delete associated snapshots
and fast-clones. Do you want to delete the virtual volume?
yes / no:
```

The default in silent mode is `yes`.

Format

`/stor/prov/vmware/vvol -id <value> delete [-async]`

Object qualifier

Qualifier	Description
`-id`	Type the ID of the virtual volume.

Action qualifier

Qualifier	Description
-async	Run the operation in asynchronous mode.

Example

The following command deletes the virtual volume with the ID naa.6006016005603c009370093e194fca3f.

```
uemcli -d 10.0.0.1 -u Local/joe -p MyPassword456! /stor/prov/vmware/
vvol –id naa.6006016005603c009370093e194fca3f delete
```

```
Virtual volume deletion will also unbind and delete associated
snapshots and fast-clones. Do you want to delete the virtual
volume?
yes / no:
yes

Storage system address: 10.0.0.1
Storage system port: 443
HTTPS connection

Operation completed successfully.
```

CHAPTER 3

Troubleshooting, Tips, and Best Practices

This chapter contains the following topics:

Troubleshooting VMware VVol datastores on Unity

This section describes possible issues and workarounds, limitations, and things to be aware of when deploying VVol datastores on the storage system. For a detailed list of system limits, refer to the *Simple Support Matrix* on the support site. For a complete list of all issues, refer to the Release Notes.

Failed to deploy VM to a VVol datastore of sufficient size

When deploying virtual volumes to VVol datastores on the storage system, the virtual volume files take up additional overhead beyond the size of the VMDK itself (data-vvol). This overhead can lead to failures when deploying new VMs to VVol datastores, even though the combined vDisk sizes are less than the overall size of the VVol datastore. This is especially true when VMs are powered on (swap-vvol) and has snapshots (memory-vvol).

For example, if the VVol datastore is 50 GB and currently has a virtual volume that is 25 GBs, attempting to deploy a new virtual volume of 20 GBs may fail due to the overhead.

It is recommended that you reserve 10-20% of the VVol datastore size as free space.

VVols inaccessible after registering a second vCenter

If a second vCenter server registers the Unity system as a VASA provider when there is already a registered vCenter, this may cause the VVol datastores to be inaccessible and thus VM operations to fail. To change vCenters, unmount all datastores and unregister the VASA provider from the original vCenter before registering the system as a VASA provider for the new vCenter.

Alternatively, to use multiple vCenters with Unity, you should deploy Platform Services Controller (PSC) as a separate appliance (refer to the VMware documentation for details: http://blogs.vmware.com/vsphere/2015/03/vcenter-server-6-topology-ha.html). You can then install multiple vCenter appliances and configure them all to use the same PSC. In this configuration, each vCenter uses the same PSC CA certificate allowing you to register Unity as the VASA Provider on multiple vCenter servers.

File VVol creation failure—Failed to create directory

When deploying a File VVol in vSphere and the VMware limit of eight maximum NFS datastore mounts is exceeded, vSphere returns a vague error message such as:

```
Cannot complete file creation operation.Operation failed,
diagnostics report: Hostsvc::osfs::CreateDirectory : Failed to
create directory new-vm1 (Cannot Create File.
```

This error message is less intuitive than the vSphere error that displays when deploying an NFS datastore that exceeds this limit: `NFS has reached the maximum number of supported volumes.`

For instructions on increasing the limit of eight maximum NFS mounts in vSphere, refer to the following VMware Knowledge Base article: https://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2239

VVols changes fail during an SP reboot

VVol changes made in vSphere may appear to have failed when Unity has an SP reboot..

Some VVol operations initiated through vSphere, such as SPBM migrations, may appear to fail if there is a concurrent SP reboot on the Unity system. This occurs because during an SP reboot, VASA is temporarily unavailable. Errors such as the following may display in vSphere:

```
The ESXi VVol session is invalid.
```

In most of these cases, the operations did complete successfully on Unity, but vSphere was not unable to get an accurate status through VASA. Occasionally, the operations also fail on Unity and error messages similar to the following may display in Unisphere:

```
Task was rolled back and marked as failed. This is because some
tasks failed or SP rebooted during task execution.
```

Restart the vSphere operation once the Unity system comes back online after the SP reboot.

VVol operations time out under high stress loads

With high-stress workloads where many VMs are created/booted in parallel, such as a bootstorm in a VDI environment, sporadic timeouts of VVols operations may occur. This is more likely to occur on arrays that use NL-SAS system drives.

Adjust the settings in vSphere to reduce the number of possible concurrent VVols operations.

VMware Certificate Authority (VMCA) support

In vSphere 6.0 and later, there are three different modes for how the Certificate Authority (CA) provisions certificates for ESXi hosts and vCenter servers:

1. Using the VMCA (default).
2. Using the VMCA as a subordinate CA to a custom certificate authority.
3. Using a custom CA as the direct root CA.

The Unity system supports only the default configuration where the VMCA provisions certificates as the root certificate authority. ESXi hosts and vCenter servers are authenticated by ensuring that the client certificate presented to the array has been signed by a trusted CA, which must be the VMCA for Unity systems.

Refer to the following VMware article for more details on CA modes for vSphere 6.0 and later:

https://pubs.vmware.com/vsphere-60/index.jsp#com.vmware.vsphere.security.doc/GUID-4D658104-1D80-441D-B6BA-4CBBCD0EDD3C.html

Note

Unity VVol datastores do not support full VVol datastore isolation between independent vSphere components using the VASA control path.

VMware Horizon support

The current Unity VVol implementation has not yet been fully certified for use with VMware Horizon View for Virtual Desktop Infrastructure (VDI). It is recommended that you use VDI and Unity when deploying less than 500 desktops.