vSphere Storage

ESXi 5.0

vCenter Server 5.0

This document supports the version of each product listed and supports all subsequent versions until the document is replaced by a new edition. To check for more recent editions of this document, see http://www.vmware.com/support/pubs.

EN-000603-00

vmware®

You can find the most up-to-date technical documentation on the VMware Web site at:

http://www.vmware.com/support/

The VMware Web site also provides the latest product updates.

If you have comments about this documentation, submit your feedback to:

docfeedback@vmware.com

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Contents

About vSphere Storage 7

1 Introduction to Storage 9
- Storage Virtualization 9
- Supported Storage Adapters 10
- Types of Physical Storage 11
- Target and Device Representations 15
- Viewing Storage Devices 16
- Displaying Datastores 19
- How Virtual Machines Access Storage 20
- Comparing Types of Storage 21

2 Overview of Using ESXi with a SAN 23
- ESXi and SAN Use Cases 24
- Specifics of Using SAN Storage with ESXi 24
- Making LUN Decisions 24
- Choosing Virtual Machine Locations 26
- Layered Applications 27
- Third-Party Management Applications 28
- SAN Storage Backup Considerations 28

3 Using ESXi with Fibre Channel SAN 31
- Fibre Channel SAN Concepts 31
- Using Zoning with Fibre Channel SANs 32
- How Virtual Machines Access Data on a Fibre Channel SAN 33

4 Configuring Fibre Channel Storage 35
- ESXi Fibre Channel SAN Requirements 35
- Installation and Setup Steps 36
- Configuring FCoE Adapters 37
- N-Port ID Virtualization 39

5 Modifying Fibre Channel Storage for ESXi 43
- Testing ESXi SAN Configurations 43
- General Setup Considerations for Fibre Channel SAN Arrays 44
- EMC CLARiiON Storage Systems 44
- EMC Symmetrix Storage Systems 45
- IBM System Storage DS4800 Storage Systems 46
- IBM Systems Storage 8000 and IBM ESS800 47
- HP StorageWorks Storage Systems 47
- Hitachi Data Systems Storage 48

Network Appliance Storage 48
LSI-Based Storage Systems 49

6 Booting ESXi from Fibre Channel SAN 51
Boot from SAN Benefits 51
Boot from Fibre Channel SAN Requirements and Considerations 52
Getting Ready for Boot from SAN 52
Configure Emulex HBA to Boot from SAN 53
Configure QLogic HBA to Boot from SAN 55

7 Best Practices for Fibre Channel Storage 57
Preventing Fibre Channel SAN Problems 57
Disable Automatic Host Registration 58
Optimizing Fibre Channel SAN Storage Performance 58
Fibre Channel SAN Configuration Checklist 59

8 Using ESXi with iSCSI SAN 61
iSCSI SAN Concepts 61
How Virtual Machines Access Data on an iSCSI SAN 66

9 Configuring iSCSI Adapters and Storage 67
ESXi iSCSI SAN Requirements 68
ESXi iSCSI SAN Restrictions 68
Setting LUN Allocations for iSCSI 68
Network Configuration and Authentication 69
Setting Up Independent Hardware iSCSI Adapters 69
Configuring Dependent Hardware iSCSI Adapters 70
Configuring Software iSCSI Adapter 72
Setting Up iSCSI Network 74
Using Jumbo Frames with iSCSI 80
Configuring Discovery Addresses for iSCSI Adapters 81
Configuring CHAP Parameters for iSCSI Adapters 82
Configuring Advanced Parameters for iSCSI 86
iSCSI Session Management 87

10 Modifying iSCSI Storage Systems for ESXi 91
Testing ESXi iSCSI SAN Configurations 91
General Considerations for iSCSI SAN Storage Systems 92
EMC CLARiiON Storage Systems 92
EMC Symmetrix Storage Systems 93
Enable HP StorageWorks MSA1510i to Communicate with ESXi 93
HP StorageWorks EVA Storage Systems 94
NetApp Storage Systems 95
Dell EqualLogic Storage Systems 95
HP StorageWorks SAN/iQ Storage Systems 95
Dell PowerVault MD3000i Storage Systems 96
iSCSI Targets in vApps 96

11 Booting from iSCSI SAN 97
General Boot from iSCSI SAN Recommendations 97
Prepare the iSCSI SAN 98
Configure Independent Hardware iSCSI Adapter for SAN Boot 98
iBFT iSCSI Boot Overview 99

12 Best Practices for iSCSI Storage 107
Preventing iSCSI SAN Problems 107
Optimizing iSCSI SAN Storage Performance 108
Checking Ethernet Switch Statistics 111
iSCSI SAN Configuration Checklist 111

13 Working with Datastores 113
Understanding VMFS Datastores 114
NFS Datastores 127
Unmount VMFS or NFS Datastores 128
Rename VMFS or NFS Datastores 129
Group VMFS or NFS Datastores 129
Handling Storage Device Disconnections 130
Creating a Diagnostic Partition 133
Set Up Dynamic Disk Mirroring 134

14 Raw Device Mapping 135
About Raw Device Mapping 135
Raw Device Mapping Characteristics 138
Create Virtual Machines with RDMs 140
Manage Paths for a Mapped Raw LUN 141

15 Solid State Disks Enablement 143
Benefits of SSD Enablement 143
Auto-Detection of SSD Devices 143
Tag Devices as SSD 144
Untag an SSD Device 145
Untag an Automatically Detected SSD Device 146
Tag Devices as Local 146
Identify SSD Devices 147
Identifying a Virtual SSD Device 148

16 VMkernel and Storage 149
Storage APIs 150

17 Understanding Multipathing and Failover 153
Failover with Fibre Channel 153
Host-Based Failover with iSCSI 154
Array-Based Failover with iSCSI 156
Path Failover and Virtual Machines 157
Managing Multiple Paths 158
VMware Multipathing Module 159

Path Scanning and Claiming 161
Managing Storage Paths and Multipathing Plug-Ins 164

18 Storage Hardware Acceleration 173
Hardware Acceleration Benefits 173
Hardware Acceleration Requirements 174
Hardware Acceleration Support Status 174
Hardware Acceleration for Block Storage Devices 174
Hardware Acceleration on NAS Devices 179
Hardware Acceleration Considerations 181

19 Storage Thin Provisioning 183
Storage Over-Subscription 183
Virtual Disk Thin Provisioning 183
Array Thin Provisioning and VMFS Datastores 186

20 Using Storage Vendor Providers 191
Vendor Providers and Storage Data Representation 191
Vendor Provider Requirements and Considerations 192
Storage Status Reporting 192
Register Vendor Providers 193
View Vendor Provider Information 193
Unregister Vendor Providers 194
Update Vendor Providers 194

21 Virtual Machine Storage Profiles 195
Understanding Storage Capabilities 195
Understanding Virtual Machine Storage Profiles 198

22 Using vmkfstools 203
vmkfstools Command Syntax 203
vmkfstools Options 204

Index 213

About vSphere Storage

vSphere Storage describes storage options available to VMware® ESXi and explains how to configure your ESXi system so that it can use and manage different types of storage. In addition, *vSphere Storage* explicitly concentrates on Fibre Channel and iSCSI storage area networks (SANs) as storage options and discusses specifics of using ESXi in Fibre Channel and iSCSI environments.

Intended Audience

This information is for experienced system administrators who are familiar with virtual machine technology, datacenter operations, and SAN storage concepts.

Introduction to Storage 1

This introduction describes available storage options for ESXi and explains how to configure your host so that it can use and manage different types of storage.

This chapter includes the following topics:

- "Storage Virtualization," on page 9
- "Supported Storage Adapters," on page 10
- "Types of Physical Storage," on page 11
- "Target and Device Representations," on page 15
- "Viewing Storage Devices," on page 16
- "Displaying Datastores," on page 19
- "How Virtual Machines Access Storage," on page 20
- "Comparing Types of Storage," on page 21

Storage Virtualization

ESXi provides host-level storage virtualization, which logically abstracts the physical storage layer from virtual machines.

An ESXi virtual machine uses a virtual disk to store its operating system, program files, and other data associated with its activities. A virtual disk is a large physical file, or a set of files, that can be copied, moved, archived, and backed up as easily as any other file. You can configure virtual machines with multiple virtual disks.

To access virtual disks, a virtual machine uses virtual SCSI controllers. These virtual controllers include BusLogic Parallel, LSI Logic Parallel, LSI Logic SAS, and VMware Paravirtual. These controllers are the only types of SCSI controllers that a virtual machine can see and access.

Each virtual disk that a virtual machine can access through one of the virtual SCSI controllers resides on a vSphere Virtual Machine File System (VMFS) datastore, an NFS-based datastore, or on a raw disk. From the standpoint of the virtual machine, each virtual disk appears as if it were a SCSI drive connected to a SCSI controller. Whether the actual physical disk device is being accessed through parallel SCSI, iSCSI, network, or Fibre Channel adapters on the host is transparent to the guest operating system and to applications running on the virtual machine.

Supported Storage Adapters

Storage adapters provide connectivity for your ESXi host to a specific storage unit or network.

ESXi supports different classes of adapters, including SCSI, iSCSI, RAID, Fibre Channel, Fibre Channel over Ethernet (FCoE), and Ethernet. ESXi accesses the adapters directly through device drivers in the VMkernel.

Depending on the type of storage you use, you might need to enable and configure a storage adapter on your host.

For information on setting up software FCoE adapters, see "Configuring FCoE Adapters," on page 37.

For information on configuring different types of iSCSI adapters, see Chapter 9, "Configuring iSCSI Adapters and Storage," on page 67.

Displaying Storage Adapters

The host uses storage adapters to access different storage devices. You can display details for the available storage adapters and review their information.

You must enable certain adapters, for example software iSCSI or FCoE, before you can view their information.

Table 1-1. Storage Adapter Information

Adapter Information	Description
Model	Model of the adapter.
Targets (Fibre Channel and SCSI)	Number of targets accessed through the adapter.
Connected Targets (iSCSI)	Number of connected targets on an iSCSI adapter.
WWN (Fibre Channel)	World Wide Name formed according to Fibre Channel standards that uniquely identifies the FC adapter.
iSCSI Name (iSCSI)	Unique name formed according to iSCSI standards that identifies the iSCSI adapter.
iSCSI Alias (iSCSI)	A friendly name used instead of the iSCSI name.
IP Address (independent hardware iSCSI)	Address assigned to the iSCSI HBA.
Devices	All storage devices or LUNs the adapter can access.
Paths	All paths the adapter uses to access storage devices.
Properties	Link that indicates that the adapter requires additional configuration. iSCSI and FCoE adapters display this link.

View Storage Adapters Information

Use the vSphere Client to display storage adapters that your host uses and to review their information.

Procedure

1 In Inventory, select **Hosts and Clusters**.

2 Select a host and click the **Configuration** tab.

3 In Hardware, select **Storage Adapters**.

4 To view details for a specific adapter, select the adapter from the Storage Adapters list.

5 To list all storage devices the adapter can access, click **Devices**.

6 To list all paths the adapter uses, click **Paths**.

Copy Storage Adapter Identifiers to the Clipboard

If your storage adapters use unique identifiers, such as an iSCSI Name or WWN, you can copy them to a clipboard directly from the vSphere Client.

Procedure

1 In Inventory, select **Hosts and Clusters**.

2 Select a host and click the **Configuration** tab.

3 In Hardware, select **Storage Adapters**.

4 Select the adapter from the Storage Adapters list.

5 In the Details panel, select the value in the name field, right-click, and select **Copy**.

Types of Physical Storage

The ESXi storage management process starts with storage space that your storage administrator preallocates on different storage systems.

ESXi supports the following types of storage:

Local Storage	Stores virtual machine files on internal or directly connected external storage disks.
Networked Storage	Stores virtual machine files on external storage disks or arrays attached to your host through a direct connection or through a high-speed network.

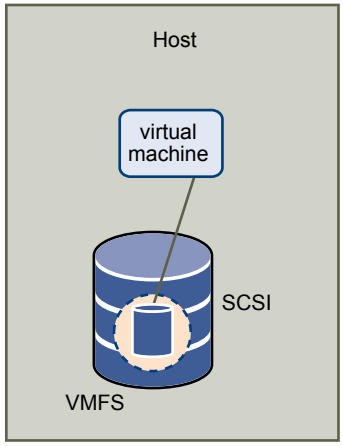
Local Storage

Local storage can be internal hard disks located inside your ESXi host, or it can be external storage systems located outside and connected to the host directly through protocols such as SAS or SATA.

Local storage does not require a storage network to communicate with your host. You need a cable connected to the storage unit and, when required, a compatible HBA in your host.

The following illustration depicts a virtual machine using local SCSI storage.

Figure 1-1. Local Storage



In this example of a local storage topology, the host uses a single connection to a storage disk. On that disk, you can create a VMFS datastore, which you use to store virtual machine disk files.

Although this storage configuration is possible, it is not a recommended topology. Using single connections between storage arrays and hosts creates single points of failure (SPOF) that can cause interruptions when a connection becomes unreliable or fails.

ESXi supports a variety of internal or external local storage devices, including SCSI, IDE, SATA, USB, and SAS storage systems. Regardless of the type of storage you use, your host hides a physical storage layer from virtual machines.

NOTE You cannot use IDE/ATA drives to store virtual machines.

Local storage devices do not support sharing across multiple hosts. A datastore on a local storage device can be accessed by only one host.

Because the majority of local storage devices do not support multiple connections, you cannot use multiple paths to access local storage.

Networked Storage

Networked storage consists of external storage systems that your ESXi host uses to store virtual machine files remotely. Typically, the host accesses these systems over a high-speed storage network.

Networked storage devices are shared. Datastores on networked storage devices can be accessed by multiple hosts concurrently. ESXi supports the following networked storage technologies.

NOTE Accessing the same storage through different transport protocols, such as iSCSI and Fibre Channel, at the same time is not supported.

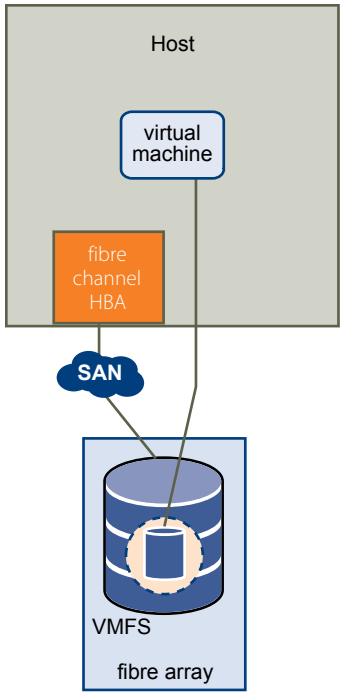
Fibre Channel (FC)

Stores virtual machine files remotely on an FC storage area network (SAN). FC SAN is a specialized high-speed network that connects your hosts to high-performance storage devices. The network uses Fibre Channel protocol to transport SCSI traffic from virtual machines to the FC SAN devices.

To connect to the FC SAN, your host should be equipped with Fibre Channel host bus adapters (HBAs). Unless you use Fibre Channel direct connect storage, you need Fibre Channel switches to route storage traffic. If your host contains FCoE (Fibre Channel over Ethernet) adapters, you can connect to your shared Fibre Channel devices by using an Ethernet network.

Fibre Channel Storage depicts virtual machines using Fibre Channel storage.

Figure 1-2. Fibre Channel Storage



In this configuration, a host connects to a SAN fabric, which consists of Fibre Channel switches and storage arrays, using a Fibre Channel adapter. LUNs from a storage array become available to the host. You can access the LUNs and create datastores for your storage needs. The datastores use the VMFS format.

For specific information on setting up the Fibre Channel SAN, see Chapter 3, "Using ESXi with Fibre Channel SAN," on page 31.

Internet SCSI (iSCSI)

Stores virtual machine files on remote iSCSI storage devices. iSCSI packages SCSI storage traffic into the TCP/IP protocol so that it can travel through standard TCP/IP networks instead of the specialized FC network. With an iSCSI connection, your host serves as the initiator that communicates with a target, located in remote iSCSI storage systems.

ESXi offers the following types of iSCSI connections:

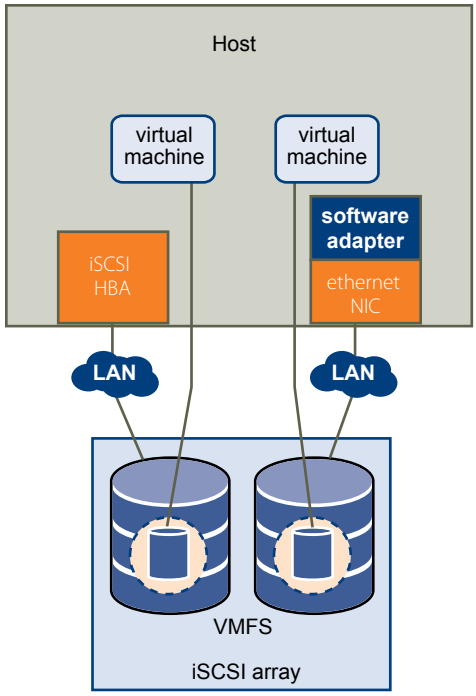
Hardware iSCSI — Your host connects to storage through a third-party adapter capable of offloading the iSCSI and network processing. Hardware adapters can be dependent and independent.

Software iSCSI — Your host uses a software-based iSCSI initiator in the VMkernel to connect to storage. With this type of iSCSI connection, your host needs only a standard network adapter for network connectivity.

You must configure iSCSI initiators for the host to access and display iSCSI storage devices.

iSCSI Storage depicts different types of iSCSI initiators.

Figure 1-3. iSCSI Storage



In the left example, the host uses the hardware iSCSI adapter to connect to the iSCSI storage system.

In the right example, the host uses a software iSCSI adapter and an Ethernet NIC to connect to the iSCSI storage.

iSCSI storage devices from the storage system become available to the host. You can access the storage devices and create VMFS datastores for your storage needs.

For specific information on setting up the iSCSI SAN, see Chapter 8, "Using ESXi with iSCSI SAN," on page 61.

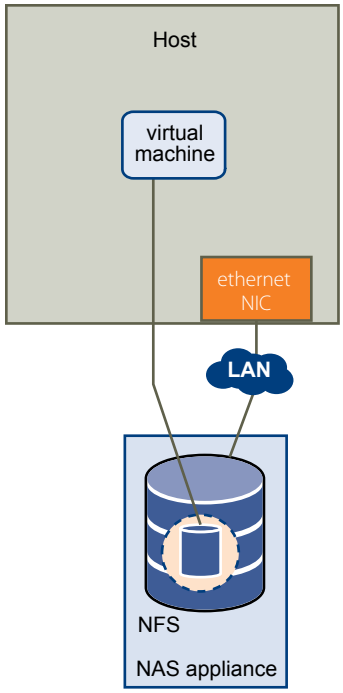
Network-attached Storage (NAS)

Stores virtual machine files on remote file servers accessed over a standard TCP/IP network. The NFS client built into ESXi uses Network File System (NFS) protocol version 3 to communicate with the NAS/NFS servers. For network connectivity, the host requires a standard network adapter.

NOTE ESXi does not support the delegate user functionality that enables access to NFS volumes using non-root credentials.

NFS Storage depicts a virtual machine using the NFS volume to store its files. In this configuration, the host connects to the NFS server, which stores the virtual disk files, through a regular network adapter.

Figure 1-4. NFS Storage



Shared Serial Attached SCSI (SAS)

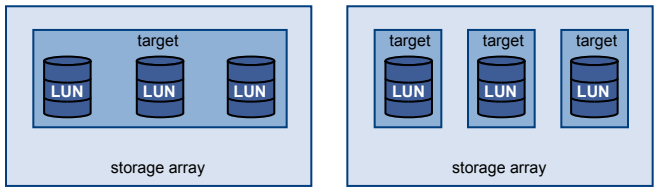
Stores virtual machines on direct-attached SAS storage systems that offer shared access to multiple hosts. This type of access permits multiple hosts to access the same VMFS datastore on a LUN.

Target and Device Representations

In the ESXi context, the term target identifies a single storage unit that the host can access. The terms device and LUN describe a logical volume that represents storage space on a target. Typically, the terms device and LUN, in the ESXi context, mean a SCSI volume presented to the host from a storage target and available for formatting.

Different storage vendors present the storage systems to ESXi hosts in different ways. Some vendors present a single target with multiple storage devices or LUNs on it, while others present multiple targets with one LUN each.

Figure 1-5. Target and LUN Representations



In this illustration, three LUNs are available in each configuration. In one case, the host sees one target, but that target has three LUNs that can be used. Each LUN represents an individual storage volume. In the other example, the host sees three different targets, each having one LUN.

Targets that are accessed through the network have unique names that are provided by the storage systems. The iSCSI targets use iSCSI names, while Fibre Channel targets use World Wide Names (WWNs).

NOTE ESXi does not support accessing the same LUN through different transport protocols, such as iSCSI and Fibre Channel.

A device, or LUN, is identified by its UUID name. If a LUN is shared by multiple hosts, it must be presented to all host with the same UUID.

Viewing Storage Devices

You can display all storage devices or LUNs available to the host, including all local and networked devices. If you use third-party multipathing plug-ins, the storage devices available through the plug-ins also appear on the list.

For each storage adapter, you can display a separate list of storage devices available for this adapter.

Generally, when you review storage devices, you see the following information.

Table 1-2. Storage Device Information

Storage Device Information	Description
Name	Also called Display Name. It is a name that the ESXi host assigns to the device based on the storage type and manufacturer. You can change this name to a name of your choice.
Identifier	A universally unique identifier that is intrinsic to the device.
Runtime Name	The name of the first path to the device.
Operational State	Indicates whether the device is mounted or unmounted. For details, see "Detach Storage Devices," on page 130.
LUN	Logical Unit Number (LUN) within the SCSI target.
Type	Type of device, for example, disk or CD-ROM.
Drive Type	Information about whether the device is a solid-state drive (SSD) or a regular non-SSD hard drive. For details, see Chapter 15, "Solid State Disks Enablement," on page 143.
Transport	Transportation protocol your host uses to access the device.
Capacity	Total capacity of the storage device.
Owner	The plug-in, such as the NMP or a third-party plug-in, that the host uses to manage paths to the storage device. For details, see "Managing Multiple Paths," on page 158.
Hardware Acceleration	Information about whether the storage device assists the host with virtual machine management operations. The status can be Supported, Not Supported, or Unknown. For details, see Chapter 18, "Storage Hardware Acceleration," on page 173.
Location	A path to the storage device in the `/vmfs/devices/` directory.
Partition Format	A partition scheme used by the storage device. It could be of a master boot record (MBR) or GUID partition table (GPT) format. The GPT devices can support datastores greater than 2TB. For more information, see "VMFS Datastores and Storage Disk Formats," on page 115.
Partitions	Primary and logical partitions, including a VMFS datastore, if configured.

Understanding Storage Device Naming

Each storage device, or LUN, is identified by several names.

Device Identifiers

Depending on the type of storage, the ESXi host uses different algorithms and conventions to generate an identifier for each storage device.

SCSI INQUIRY identifiers. The host uses the SCSI INQUIRY command to query a storage device and uses the resulting data, in particular the Page 83 information, to generate a unique identifier. Device identifiers that are based on Page 83 are unique across all hosts, persistent, and have one of the following formats:

- naa.*number*
- t10.*number*
- eui.*number*

These formats follow the T10 committee standards. See the SCSI-3 documentation on the T10 committee Web site.

Path-based identifier. When the device does not provide the Page 83 information, the host generates an mpx.*path* name, where *path* represents the path to the device, for example, mpx.vmhba1:C0:T1:L3. This identifier can be used in the same way as the SCSI INQUIRY identifies.

The mpx. identifier is created for local devices on the assumption that their path names are unique. However, this identifier is neither unique nor persistent and could change after every boot.

Legacy Identifier

In addition to the SCSI INQUIRY or mpx. identifiers, for each device, ESXi generates an alternative legacy name. The identifier has the following format:

vml.*number*

The legacy identifier includes a series of digits that are unique to the device and can be derived in part from the Page 83 information, if it is available. For nonlocal devices that do not support Page 83 information, the vml. name is used as the only available unique identifier.

Example: Displaying Device Names in the vSphere CLI

You can use the **esxcli --server=*server_name* storage core device list** command to display all device names in the vSphere CLI. The output is similar to the following example:

```
# esxcli --server=server_name storage core device list
naa.number
    Display Name: DGC Fibre Channel Disk(naa.number)
    ...
    Other UIDs:vml.number
```

Runtime Name

In the vSphere Client, you can see the device identifier and a runtime name. The runtime name is generated by the host and represents the name of the first path to the device. It is not a reliable identifier for the device, and is not persistent.

Typically, the path to the device has the following format:

vmhba*Adapter*:C*Channel*:T*Target*:L*LUN*

- vmhba*Adapter* is the name of the storage adapter. The name refers to the physical adapter on the host, not to the SCSI controller used by the virtual machines.
- C*Channel* is the storage channel number.

  Software iSCSI adapters and dependent hardware adapters use the channel number to show multiple paths to the same target.
- T*Target* is the target number. Target numbering is determined by the host and might change if the mappings of targets visible to the host change. Targets that are shared by different hosts might not have the same target number.
- L*LUN* is the LUN number that shows the position of the LUN within the target. The LUN number is provided by the storage system. If a target has only one LUN, the LUN number is always zero (0).

For example, `vmhba1:C0:T3:L1` represents LUN1 on target 3 accessed through the storage adapter vmhba1 and channel 0.

Display Storage Devices for a Host

Use the vSphere Client to display all storage devices or LUNs available to a host. If you use any third-party multipathing plug-ins, the storage devices available through the plug-ins also appear on the list.

Procedure

1 In Inventory, select **Hosts and Clusters**.

2 Select a host and click the **Configuration** tab.

3 In Hardware, select **Storage**.

4 Click **Devices**.

5 To view additional details about a specific device, select the device from the list.

Display Storage Devices for an Adapter

Use the vSphere Client to display a list of storage devices accessible to a specific storage adapter on the host.

Procedure

1 In Inventory, select **Hosts and Clusters**.

2 Select a host and click the **Configuration** tab.

3 In Hardware, select **Storage Adapters**.

4 Select the adapter from the Storage Adapters list.

5 Click **Devices**.

Copy Storage Device Identifiers to the Clipboard

A storage device identifier is a universally unique ID assigned to a storage device or LUN. Depending on the type of storage, different algorithms are used to create the identifier and it can be long and complex. You can copy the storage device identifier directly from the vSphere Client.

Procedure

1 Display a list of storage devices.

2 Right-click a device and select **Copy identifier to clipboard**.

Displaying Datastores

Datastores are logical containers, analogous to file systems, that hide specifics of each storage device and provide a uniform model for storing virtual machine files. You can display all datastores available to your hosts and analyze their properties.

Datastores are added to the vSphere Client in the following ways:

- Created on an available storage device.
- Discovered when a host is added to the inventory. When you add a host to the inventory, the vSphere Client displays any datastores available to the host.

If your vSphere Client is connected to a vCenter Server system, you can see datastore information in the Datastores and Datastore Clusters view. This view displays all datastores and datastore clusters in the inventory, arranged by a datacenter. Through this view, you can organize datastores into folder hierarchies, create datastores, edit their properties, or remove existing datastores.

This view is comprehensive and shows all information for your datastores and clusters including hosts and virtual machines using the datastores, storage reporting information, permissions, alarms, tasks and events, storage topology, and storage reports.

NOTE The Datastores and Datastore Clusters view is not available when the vSphere Client connects directly to your host. In this case, review datastore information through the host storage configuration tab.

The following table describes the datastore details that you can see when you review datastores.

Table 1-3. Datastore Information

Datastore Information	Description
Identification	Editable name that you assign to the datastore.
Device	Storage device on which the datastore is deployed.
Drive Type	Type of underlying storage device, a Solid State Drive (SSD) or a regular non-SSD hard drive. For details, see Chapter 15, "Solid State Disks Enablement," on page 143.
Capacity	Total formatted capacity of the datastore.
Free	Available space.
Type	File system that the datastore uses, either VMFS or NFS. For information about datastores and how to upgrade to VMFS5, see Chapter 13, "Working with Datastores," on page 113.
Storage I/O Control	Information on whether cluster-wide storage I/O prioritization is enabled. See the *vSphere Resource Management* documentation.
Hardware Acceleration	Information on whether the underlying storage device supports hardware acceleration. The status can be Supported, Not Supported, or Unknown. For details, see Chapter 18, "Storage Hardware Acceleration," on page 173.
Location (VMFS datastores)	A path to the datastore in the `/vmfs/volumes/` directory.
Server (NFS datastores)	Name or IP address of a NAS server.
Folder (NFS datastores)	Name of a mounted folder.
Extents (VMFS datastores)	Individual extents that the datastore spans and their capacity.
System Storage Capability	Storage capabilities reported by supported storage devices and inherited by the datastores. You cannot modify them.
User-defined Storage Capability	Storage capabilities that you define and associate with datastores. For information, see "Understanding Storage Capabilities," on page 195.

Table 1-3. Datastore Information (Continued)

Datastore Information	Description
Path Selection (VMFS datastores)	Path selection policy the host uses to access storage. For more information, see Chapter 17, "Understanding Multipathing and Failover," on page 153.
Paths (VMFS datastores)	Number of paths used to access storage and their status.

Review Datastore Properties

Use the vSphere Client to display all datastores available to the hosts and analyze their properties.

Procedure

1 In Inventory, select **Hosts and Clusters**.

2 Select a host and click the **Configuration** tab.

3 In Hardware, select **Storage**.

4 Click the **Datastores** view.

5 To display details for a particular datastore, select the datastore from the list.

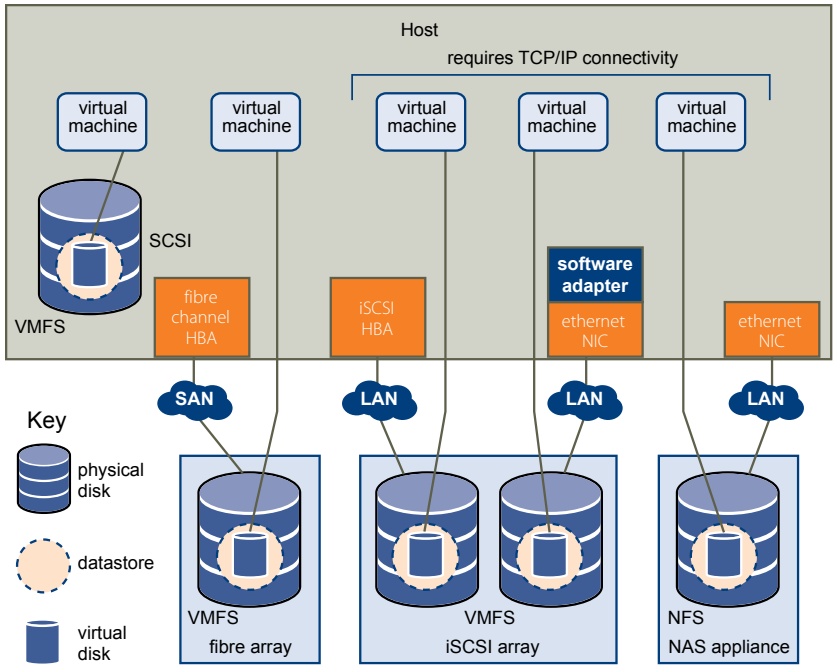
How Virtual Machines Access Storage

When a virtual machine communicates with its virtual disk stored on a datastore, it issues SCSI commands. Because datastores can exist on various types of physical storage, these commands are encapsulated into other forms, depending on the protocol that the ESXi host uses to connect to a storage device.

ESXi supports Fibre Channel (FC), Internet SCSI (iSCSI), Fibre Channel over Ethernet (FCoE), and NFS protocols. Regardless of the type of storage device your host uses, the virtual disk always appears to the virtual machine as a mounted SCSI device. The virtual disk hides a physical storage layer from the virtual machine's operating system. This allows you to run operating systems that are not certified for specific storage equipment, such as SAN, inside the virtual machine.

The following graphic depicts five virtual machines using different types of storage to illustrate the differences between each type.

Figure 1-6. Virtual machines accessing different types of storage



NOTE This diagram is for conceptual purposes only. It is not a recommended configuration.

You can use maps on the **Storage Views** tab to visually represent and understand the relationships between virtual machines on your host and all available virtual and physical storage resources. For more information, see the *vSphere Monitoring and Performance* documentation.

Comparing Types of Storage

Whether certain vSphere functionality is supported might depend on the storage technology that you use.

The following table compares networked storage technologies that ESXi supports.

Table 1-4. Networked Storage that ESXi Supports

Technology	Protocols	Transfers	Interface
Fibre Channel	FC/SCSI	Block access of data/LUN	FC HBA
Fibre Channel over Ethernet	FCoE/SCSI	Block access of data/LUN	■ Converged Network Adapter (hardware FCoE) ■ NIC with FCoE support (software FCoE)
iSCSI	IP/SCSI	Block access of data/LUN	■ iSCSI HBA or iSCSI-enabled NIC (hardware iSCSI) ■ Network adapter (software iSCSI)
NAS	IP/NFS	File (no direct LUN access)	Network adapter

The following table compares the vSphere features that different types of storage support.

Table 1-5. vSphere Features Supported by Storage

Storage Type	Boot VM	vMotion	Datastore	RDM	VM Cluster	VMware HA and DRS	Storage APIs - Data Protection
Local Storage	Yes	No	VMFS	No	Yes	No	Yes
Fibre Channel	Yes	Yes	VMFS	Yes	Yes	Yes	Yes
iSCSI	Yes	Yes	VMFS	Yes	No	Yes	Yes
NAS over NFS	Yes	Yes	NFS	No	No	Yes	Yes

NOTE Local storage supports a cluster of virtual machines on a single host (also known as a cluster in a box). A shared virtual disk is required. For more information about this configuration, see the *vSphere Resource Management* documentation.

Overview of Using ESXi with a SAN 2

Using ESXi with a SAN improves flexibility, efficiency, and reliability. Using ESXi with a SAN also supports centralized management, failover, and load balancing technologies.

The following are benefits of using ESXi with a SAN:

- You can store data securely and configure multiple paths to your storage, eliminating a single point of failure.
- Using a SAN with ESXi systems extends failure resistance to the server. When you use SAN storage, all applications can instantly be restarted on another host after the failure of the original host.
- You can perform live migration of virtual machines using VMware vMotion.
- Use VMware High Availability (HA) in conjunction with a SAN to restart virtual machines in their last known state on a different server if their host fails.
- Use VMware Fault Tolerance (FT) to replicate protected virtual machines on two different hosts. Virtual machines continue to function without interruption on the secondary host if the primary one fails.
- Use VMware Distributed Resource Scheduler (DRS) to migrate virtual machines from one host to another for load balancing. Because storage is on a shared SAN array, applications continue running seamlessly.
- If you use VMware DRS clusters, put an ESXi host into maintenance mode to have the system migrate all running virtual machines to other ESXi hosts. You can then perform upgrades or other maintenance operations on the original host.

The portability and encapsulation of VMware virtual machines complements the shared nature of this storage. When virtual machines are located on SAN-based storage, you can quickly shut down a virtual machine on one server and power it up on another server, or suspend it on one server and resume operation on another server on the same network. This ability allows you to migrate computing resources while maintaining consistent shared access.

This chapter includes the following topics:

- "ESXi and SAN Use Cases," on page 24
- "Specifics of Using SAN Storage with ESXi," on page 24
- "Making LUN Decisions," on page 24
- "Choosing Virtual Machine Locations," on page 26
- "Layered Applications," on page 27
- "Third-Party Management Applications," on page 28
- "SAN Storage Backup Considerations," on page 28

ESXi and SAN Use Cases

When used with a SAN, ESXi can benefit from multiple vSphere features, including Storage vMotion, Distributed Resource Scheduler (DRS), High Availability, and so on.

Using ESXi in conjunction with a SAN is effective for the following tasks:

Storage consolidation and simplification of storage layout
If you are working with multiple hosts, and each host is running multiple virtual machines, the storage on the hosts is no longer sufficient and external storage is required. Choose a SAN for external storage to provide a simpler system architecture along with other benefits.

Maintenance with zero downtime
When performing ESXi host or infrastructure maintenance, use vMotion to migrate virtual machines to other host. If shared storage is on the SAN, you can perform maintenance without interruptions to the users of the virtual machines. Virtual machine working processes continue throughout a migration.

Load balancing
You can add a host to a DRS cluster, and the host's resources become part of the cluster's resources. The distribution and usage of CPU and memory resources for all hosts and virtual machines in the cluster are continuously monitored. DRS compares these metrics to an ideal resource utilization. Ideal utilization takes into account the attributes of the cluster's resource pools and virtual machines, the current demand, and the imbalance target. It then performs (or recommends) virtual machine migrations accordingly.

Disaster recovery
You can use VMware High Availability to configure multiple ESXi hosts as a cluster to provide rapid recovery from outages and cost-effective high availability for applications running in virtual machines.

Simplified array migrations and storage upgrades
When you purchase new storage systems or arrays, use Storage vMotion to perform live automated migration of virtual machine disk files from existing storage to their new destination without interruptions to the users of the virtual machines.

Specifics of Using SAN Storage with ESXi

Using a SAN in conjunction with an ESXi host differs from traditional SAN usage in a variety of ways.

When you use SAN storage with ESXi, keep in mind the following considerations:

- You cannot directly access the virtual machine operating system that uses the storage. With traditional tools, you can monitor only the VMware ESXi operating system. You use the vSphere Client to monitor virtual machines.
- The HBA visible to the SAN administration tools is part of the ESXi system, not part of the virtual machine.
- Your ESXi system performs multipathing for you.

Making LUN Decisions

You must plan how to set up storage for your ESXi systems before you format LUNs with VMFS datastores.

When you make your LUN decision, keep in mind the following considerations:

- Each LUN should have the correct RAID level and storage characteristic for the applications running in virtual machines that use the LUN.
- Each LUN must contain only one VMFS datastore.

- If multiple virtual machines access the same VMFS, use disk shares to prioritize virtual machines. See "Use Disk Shares to Prioritize Virtual Machines," on page 26.

You might want fewer, larger LUNs for the following reasons:

- More flexibility to create virtual machines without asking the storage administrator for more space.
- More flexibility for resizing virtual disks, doing snapshots, and so on.
- Fewer VMFS datastores to manage.

You might want more, smaller LUNs for the following reasons:

- Less wasted storage space.
- Different applications might need different RAID characteristics.
- More flexibility, as the multipathing policy and disk shares are set per LUN.
- Use of Microsoft Cluster Service requires that each cluster disk resource is in its own LUN.
- Better performance because there is less contention for a single volume.

When the storage characterization for a virtual machine is not available, there is often no simple method to determine the number and size of LUNs to provision. You can experiment using either a predictive or adaptive scheme.

Use the Predictive Scheme to Make LUN Decisions

When setting up storage for ESXi systems, before creating VMFS datastores, you must decide on the size and number of LUNs to provision. You can experiment using the predictive scheme.

Procedure

1 Provision several LUNs with different storage characteristics.

2 Create a VMFS datastore on each LUN, labeling each datastore according to its characteristics.

3 Create virtual disks to contain the data for virtual machine applications in the VMFS datastores created on LUNs with the appropriate RAID level for the applications' requirements.

4 Use disk shares to distinguish high-priority from low-priority virtual machines.

NOTE Disk shares are relevant only within a given host. The shares assigned to virtual machines on one host have no effect on virtual machines on other hosts.

5 Run the applications to determine whether virtual machine performance is acceptable.

Use the Adaptive Scheme to Make LUN Decisions

When setting up storage for ESXi hosts, before creating VMFS datastores, you must decide on the number and size of LUNS to provision. You can experiment using the adaptive scheme.

Procedure

1 Provision a large LUN (RAID 1+0 or RAID 5), with write caching enabled.

2 Create a VMFS on that LUN.

3 Create four or five virtual disks on the VMFS.

4 Run the applications to determine whether disk performance is acceptable.

If performance is acceptable, you can place additional virtual disks on the VMFS. If performance is not acceptable, create a new, large LUN, possibly with a different RAID level, and repeat the process. Use migration so that you do not lose virtual machines data when you recreate the LUN.

Use Disk Shares to Prioritize Virtual Machines

If multiple virtual machines access the same VMFS datastore (and therefore the same LUN), use disk shares to prioritize the disk accesses from the virtual machines. Disk shares distinguish high-priority from low-priority virtual machines.

Procedure

1 Start a vSphere Client and connect to the vCenter Server.

2 Select the virtual machine in the inventory panel and click **Edit virtual machine settings** from the menu.

3 Click the **Resources** tab and click **Disk**.

4 Double-click the **Shares** column for the disk to modify and select the required value from the drop-down menu.

Shares is a value that represents the relative metric for controlling disk bandwidth to all virtual machines. The values Low, Normal, High, and Custom are compared to the sum of all shares of all virtual machines on the host. Share allocation symbolic values can be used to configure their conversion into numeric values.

5 Click **OK** to save your selection.

NOTE Disk shares are relevant only within a given ESXi host. The shares assigned to virtual machines on one host have no effect on virtual machines on other hosts.

Choosing Virtual Machine Locations

When you're working on optimizing performance for your virtual machines, storage location is an important factor. A trade-off always exists between expensive storage that offers high performance and high availability and storage with lower cost and lower performance.

Storage can be divided into different tiers depending on a number of factors:

- High Tier. Offers high performance and high availability. Might offer built-in snapshots to facilitate backups and point-in-time (PiT) restorations. Supports replication, full SP redundancy, and SAS drives. Uses high-cost spindles.
- Mid Tier. Offers mid-range performance, lower availability, some SP redundancy, and SCSI or SAS drives. May offer snapshots. Uses medium-cost spindles.
- Lower Tier. Offers low performance, little internal storage redundancy. Uses low end SCSI drives or SATA (serial low-cost spindles).

Not all applications need to be on the highest-performance, most-available storage—at least not throughout their entire life cycle.

NOTE If you need some of the functionality of the high tier, such as snapshots, but do not want to pay for it, you might be able to achieve some of the high-performance characteristics in software. For example, you can create snapshots in software.

When you decide where to place a virtual machine, ask yourself these questions:

- How critical is the virtual machine?
- What are its performance and availability requirements?
- What are its PiT restoration requirements?
- What are its backup requirements?
- What are its replication requirements?

A virtual machine might change tiers throughout its life cycle because of changes in criticality or changes in technology that push higher-tier features to a lower tier. Criticality is relative and might change for a variety of reasons, including changes in the organization, operational processes, regulatory requirements, disaster planning, and so on.

Layered Applications

SAN administrators customarily use specialized array-based software for backup, disaster recovery, data mining, forensics, and configuration testing.

Storage providers typically supply two types of advanced services for their LUNs: snapshotting and replication.

- Snapshotting creates space with efficient copies of LUNs that share common blocks of data. In general, snapshotting is used locally on the same storage systems as the primary LUN for quick backups, application testing, forensics, or data mining.
- Replication creates full copies of LUNs. Replicas are usually made to separate storage systems, possibly separate sites to protect against major outages that incapacitate or destroy an entire array or site.

When you use an ESXi system in conjunction with a SAN, you must decide whether array-based or host-based tools are more suitable for your particular situation.

Array-Based (Third-Party) Solution

When you use an ESXi system in conjunction with a SAN, you must decide whether array-based tools are more suitable for your particular situation.

When you consider an array-based solution, keep in mind the following points:

- Array-based solutions usually result in more comprehensive statistics. With RDMs, data always takes the same path, which results in easier performance management.
- Security is more transparent to the storage administrator when you use an RDM and an array-based solution because with RDMs, virtual machines more closely resemble physical machines.
- If you use an array-based solution, physical compatibility RDMs are often used for the storage of virtual machines. If you do not intend to use RDMs, check the storage vendor documentation to see if operations on LUNs with VMFS volumes are supported. If you use array operations on VMFS LUNs, carefully read the section on resignaturing.

File-Based (VMFS) Solution

When you use an ESXi system in conjunction with a SAN, you must decide whether file-based tools are more suitable for your particular situation.

When you consider a file-based solution that uses VMware tools and VMFS instead of the array tools, be aware of the following points:

- Using VMware tools and VMFS is better for provisioning. One large LUN is allocated and multiple `.vmdk` files can be placed on that LUN. With an RDM, a new LUN is required for each virtual machine.
- Snapshotting is included with your ESXi host at no extra cost.
- Using VMFS is easier for ESXi administrators.
- ESXi administrators who use the file-based solution are more independent from the SAN administrator.

Third-Party Management Applications

You can use third-party management applications in conjunction with your ESXi host.

Most SAN hardware is packaged with storage management software. In many cases, this software is a web application that can be used with any web browser connected to your network. In other cases, this software typically runs on the storage system or on a single server, independent of the servers that use the SAN for storage.

Use this third-party management software for the following tasks:

- Storage array management, including LUN creation, array cache management, LUN mapping, and LUN security.
- Setting up replication, check points, snapshots, or mirroring.

If you decide to run the SAN management software on a virtual machine, you gain the benefits of running a virtual machine, including failover using vMotion and VMware HA. Because of the additional level of indirection, however, the management software might not be able to see the SAN. In this case, you can use an RDM.

NOTE Whether a virtual machine can run management software successfully depends on the particular storage system.

SAN Storage Backup Considerations

Having a proper backup strategy is one of the most important aspects of SAN management. In the SAN environment, backups have two goals. The first goal is to archive online data to offline media. This process is repeated periodically for all online data on a time schedule. The second goal is to provide access to offline data for recovery from a problem. For example, database recovery often requires retrieval of archived log files that are not currently online.

Scheduling a backup depends on a number of factors:

- Identification of critical applications that require more frequent backup cycles within a given period of time.
- Recovery point and recovery time goals. Consider how precise your recovery point needs to be, and how long you are willing to wait for it.
- The rate of change (RoC) associated with the data. For example, if you are using synchronous/asynchronous replication, the RoC affects the amount of bandwidth required between the primary and secondary storage devices.
- Overall impact on SAN environment, storage performance (while backing up), and other applications.
- Identification of peak traffic periods on the SAN (backups scheduled during those peak periods can slow the applications and the backup process).
- Time to schedule all backups within the datacenter.
- Time it takes to back up an individual application.
- Resource availability for archiving data; usually offline media access (tape).

Include a recovery-time objective for each application when you design your backup strategy. That is, consider the time and resources necessary to perform a backup. For example, if a scheduled backup stores so much data that recovery requires a considerable amount of time, examine the scheduled backup. Perform the backup more frequently, so that less data is backed up at a time and the recovery time decreases.

If a particular application requires recovery within a certain time frame, the backup process needs to provide a time schedule and specific data processing to meet this requirement. Fast recovery can require the use of recovery volumes that reside on online storage to minimize or eliminate the need to access slow offline media for missing data components.

Using Third-Party Backup Packages

You can use third-party backup solutions to protect system, application, and user data in your virtual machines.

VMware offers the Storage APIs - Data Protection to work in conjunction with third-party products. When using the APIs, third-party software can perform backups without loading ESXi hosts with the processing of backup tasks.

The third-party products using the Storage APIs - Data Protection can perform the following backup tasks:

- Perform full, differential, and incremental image backup and restore of virtual machines.
- Perform file-level backup of virtual machines that use supported Windows and Linux operating systems.
- Ensure data consistency by using Microsoft Volume Shadow Copy Services (VSS) for virtual machines that run supported Microsoft Windows operating systems.

Because the Storage APIs - Data Protection leverage the snapshot capabilities of VMFS, backups that you can perform do not require downtime for virtual machines. These backups are nondisruptive, can be performed at any time, and do not need extended backup windows.

For information about the Storage APIs - Data Protection and integration with backup products, see the VMware Web site or contact you backup vendor.

Using ESXi with Fibre Channel SAN 3

When you set up ESXi hosts to use FC SAN storage arrays, special considerations are necessary. This section provides introductory information about how to use ESXi with a FC SAN array.

This chapter includes the following topics:

- "Fibre Channel SAN Concepts," on page 31
- "Using Zoning with Fibre Channel SANs," on page 32
- "How Virtual Machines Access Data on a Fibre Channel SAN," on page 33

Fibre Channel SAN Concepts

If you are an ESXi administrator planning to set up hosts to work with SANs, you must have a working knowledge of SAN concepts. You can find information about SANs in print and on the Internet. Because this industry changes constantly, check these resources frequently.

If you are new to SAN technology, familiarize yourself with the basic terminology.

A storage area network (SAN) is a specialized high-speed network that connects computer systems, or host servers, to high performance storage subsystems. The SAN components include host bus adapters (HBAs) in the host servers, switches that help route storage traffic, cables, storage processors (SPs), and storage disk arrays.

A SAN topology with at least one switch present on the network forms a SAN fabric.

To transfer traffic from host servers to shared storage, the SAN uses the Fibre Channel (FC) protocol that packages SCSI commands into Fibre Channel frames.

To restrict server access to storage arrays not allocated to that server, the SAN uses zoning. Typically, zones are created for each group of servers that access a shared group of storage devices and LUNs. Zones define which HBAs can connect to which SPs. Devices outside a zone are not visible to the devices inside the zone.

Zoning is similar to LUN masking, which is commonly used for permission management. LUN masking is a process that makes a LUN available to some hosts and unavailable to other hosts.

When transferring data between the host server and storage, the SAN uses a technique known as multipathing. Multipathing allows you to have more than one physical path from the ESXi host to a LUN on a storage system.

Generally, a single path from a host to a LUN consists of an HBA, switch ports, connecting cables, and the storage controller port. If any component of the path fails, the host selects another available path for I/O. The process of detecting a failed path and switching to another is called path failover.

Ports in Fibre Channel SAN

In the context of this document, a port is the connection from a device into the SAN. Each node in the SAN, such as a host, a storage device, or a fabric component has one or more ports that connect it to the SAN. Ports are identified in a number of ways.

WWPN (World Wide Port Name) A globally unique identifier for a port that allows certain applications to access the port. The FC switches discover the WWPN of a device or host and assign a port address to the device.

Port_ID (or port address) Within a SAN, each port has a unique port ID that serves as the FC address for the port. This unique ID enables routing of data through the SAN to that port. The FC switches assign the port ID when the device logs in to the fabric. The port ID is valid only while the device is logged on.

When N-Port ID Virtualization (NPIV) is used, a single FC HBA port (N-port) can register with the fabric by using several WWPNs. This method allows an N-port to claim multiple fabric addresses, each of which appears as a unique entity. When ESXi hosts use a SAN, these multiple, unique identifiers allow the assignment of WWNs to individual virtual machines as part of their configuration.

Fibre Channel Storage Array Types

ESXi supports different storage systems and arrays.

The types of storage that your host supports include active-active, active-passive, and ALUA-compliant.

Active-active storage system Allows access to the LUNs simultaneously through all the storage ports that are available without significant performance degradation. All the paths are active at all times, unless a path fails.

Active-passive storage system A system in which one storage processor is actively providing access to a given LUN. The other processors act as backup for the LUN and can be actively providing access to other LUNs. I/O can be successfully sent only to an active port for a given LUN. If access through the active storage port fails, one of the passive storage processors can be activated by the servers accessing it.

Asymmetrical storage system Supports Asymmetric Logical Unit Access (ALUA). ALUA-complaint storage systems provide different levels of access per port. ALUA allows hosts to determine the states of target ports and prioritize paths. The host uses some of the active paths as primary while others as secondary.

Using Zoning with Fibre Channel SANs

Zoning provides access control in the SAN topology. Zoning defines which HBAs can connect to which targets. When you configure a SAN by using zoning, the devices outside a zone are not visible to the devices inside the zone.

Zoning has the following effects:

- Reduces the number of targets and LUNs presented to a host.
- Controls and isolates paths in a fabric.
- Can prevent non-ESXi systems from accessing a particular storage system, and from possibly destroying VMFS data.
- Can be used to separate different environments, for example, a test from a production environment.

With ESXi hosts, use a single-initiator zoning or a single-initiator-single-target zoning. The latter is a preferred zoning practice. Using the more restrictive zoning prevents problems and misconfigurations that can occur on the SAN.

For detailed instructions and best zoning practices, contact storage array or switch vendors.

How Virtual Machines Access Data on a Fibre Channel SAN

ESXi stores a virtual machine's disk files within a VMFS datastore that resides on a SAN storage device. When virtual machine guest operating systems issue SCSI commands to their virtual disks, the SCSI virtualization layer translates these commands to VMFS file operations.

When a virtual machine interacts with its virtual disk stored on a SAN, the following process takes place:

1 When the guest operating system in a virtual machine reads or writes to a SCSI disk, it issues SCSI commands to the virtual disk.

2 Device drivers in the virtual machine’s operating system communicate with the virtual SCSI controllers.

3 The virtual SCSI controller forwards the command to the VMkernel.

4 The VMkernel performs the following tasks.

   a Locates the file in the VMFS volume that corresponds to the guest virtual machine disk.

   b Maps the requests for the blocks on the virtual disk to blocks on the appropriate physical device.

   c Sends the modified I/O request from the device driver in the VMkernel to the physical HBA.

5 The physical HBA performs the following tasks.

   a Packages the I/O request according to the rules of the FC protocol.

   b Transmits the request to the SAN.

6 Depending on a port the HBA uses to connect to the fabric, one of the SAN switches receives the request and routes it to the storage device that the host wants to access.

Configuring Fibre Channel Storage 4

When you use ESXi systems with SAN storage, specific hardware and system requirements exist.

This chapter includes the following topics:

- "ESXi Fibre Channel SAN Requirements," on page 35
- "Installation and Setup Steps," on page 36
- "Configuring FCoE Adapters," on page 37
- "N-Port ID Virtualization," on page 39

ESXi Fibre Channel SAN Requirements

In preparation for configuring your SAN and setting up your ESXi system to use SAN storage, review the requirements and recommendations.

- Make sure that the SAN storage hardware and firmware combinations you use are supported in conjunction with ESXi systems. For an up-to-date list, see the *vSphere Compatibility Guide*.
- Configure your system to have only one VMFS volume per LUN.
- Unless you are using diskless servers, do not set up the diagnostic partition on a SAN LUN.

  In the case of diskless servers that boot from a SAN, a shared diagnostic partition is appropriate.
- Use RDMs to access raw disks. For information, see Chapter 14, "Raw Device Mapping," on page 135.
- For multipathing to work properly, each LUN must present the same LUN ID number to all ESXi hosts.
- Make sure the storage device driver specifies a large enough queue. You can set the queue depth for the physical HBA during system setup. For information on changing queue depth for HBAs and virtual machines, see the *vSphere Troubleshooting* documentation.
- On virtual machines running Microsoft Windows, increase the value of the SCSI `TimeoutValue` parameter to 60. This increase allows Windows to better tolerate delayed I/O resulting from path failover. For information, see "Set Timeout on Windows Guest OS," on page 157.

ESXi Fibre Channel SAN Restrictions

When you use ESXi with a SAN, certain restrictions apply.

- ESXi does not support FC connected tape devices.
- You cannot use virtual machine multipathing software to perform I/O load balancing to a single physical LUN.

- You cannot use multipathing software inside a virtual machine to perform I/O load balancing to a single physical LUN. However, when your Microsoft Windows virtual machine uses dynamic disks, this restriction does not apply. For information about configuring dynamic disks, see "Set Up Dynamic Disk Mirroring," on page 134.

Setting LUN Allocations

This topic provides general information about how to allocate LUNs when your ESXi works in conjunction with SAN.

When you set LUN allocations, be aware of the following points:

Storage provisioning

To ensure that the ESXi system recognizes the LUNs at startup time, provision all LUNs to the appropriate HBAs before you connect the SAN to the ESXi system.

VMware recommends that you provision all LUNs to all ESXi HBAs at the same time. HBA failover works only if all HBAs see the same LUNs.

For LUNs that will be shared among multiple hosts, make sure that LUN IDs are consistent across all hosts. For example, LUN 5 should be mapped to host 1, host 2, and host 3 as LUN 5.

vMotion and VMware DRS

When you use vCenter Server and vMotion or DRS, make sure that the LUNs for the virtual machines are provisioned to all ESXi hosts. This provides the most ability to move virtual machines.

Active-active compared to active-passive arrays

When you use vMotion or DRS with an active-passive SAN storage device, make sure that all ESXi systems have consistent paths to all storage processors. Not doing so can cause path thrashing when a vMotion migration occurs.

For active-passive storage arrays not listed in Storage/SAN Compatibility, VMware does not support storage port failover. In those cases, you must connect the server to the active port on the storage array. This configuration ensures that the LUNs are presented to the ESXi host.

Setting Fibre Channel HBAs

Typically, FC HBAs that you use on your ESXi host work correctly with the default configuration settings.

You should follow the configuration guidelines provided by your storage array vendor. During FC HBA setup, consider the following issues.

- Do not mix FC HBAs from different vendors in a single host. Having different models of the same HBA is supported, but a single LUN cannot be accessed through two different HBA types, only through the same type.
- Ensure that the firmware level on each HBA is the same.
- Set the timeout value for detecting a failover. To ensure optimal performance, do not change the default value.

Installation and Setup Steps

This topic provides an overview of installation and setup steps that you need to follow when configuring your SAN environment to work with ESXi.

Follow these steps to configure your ESXi SAN environment.

1 Design your SAN if it is not already configured. Most existing SANs require only minor modification to work with ESXi.

2 Check that all SAN components meet requirements.

3 Perform any necessary storage array modification.

Most vendors have vendor-specific documentation for setting up a SAN to work with VMware ESXi.

4 Set up the HBAs for the hosts you have connected to the SAN.

5 Install ESXi on the hosts.

6 Create virtual machines and install guest operating systems.

7 (Optional) Set up your system for VMware HA failover or for using Microsoft Clustering Services.

8 Upgrade or modify your environment as needed.

Configuring FCoE Adapters

ESXi can use Fibre Channel over Ethernet (FCoE) adapters to access Fibre Channel storage.

The FCoE protocol encapsulates Fibre Channel frames into Ethernet frames. As a result, your host does not need special Fibre Channel links to connect to Fibre Channel storage, but can use 10Gbit lossless Ethernet to deliver Fibre Channel traffic.

To use FCoE, you need to install FCoE adapters. The adapters that VMware supports generally fall into two categories, hardware FCoE adapters and software FCoE adapters that use the native FCoE stack in ESXi.

Hardware FCoE Adapters

This category includes completely offloaded specialized Converged Network Adapters (CNAs) that contain network and Fibre Channel functionalities on the same card.

When such adapter is installed, your host detects and can use both CNA components. In the vSphere Client, the networking component appears as a standard network adapter (vmnic) and the Fibre Channel component as a FCoE adapter (vmhba). You do not need to configure the hardware FCoE adapter to be able to use it.

Software FCoE Adapters

A software FCoE adapter uses the native FCoE protocol stack in ESXi for the protocol processing. The software FCoE adapter is used with a NIC that offers Data Center Bridging (DCB) and I/O offload capabilities. For information on NICs supporting software FCoE, see the *vSphere Compatibility Guide*.

For the software FCoE adapter, you must properly configure networking and then activate the adapter.

Note The number of software FCoE adapters you activate corresponds to the number of physical NIC ports. ESXi 5.0 supports a maximum of four software FCoE adapters on one host.

Configuration Guidelines for Software FCoE

When setting up your network environment to work with ESXi software FCoE, follow the guidelines and best practices that VMware offers.

Network Switch Guidelines

Follow these guidelines when you configure a network switch for software FCoE environment:

- On the ports that communicate with your ESXi host, disable the Spanning Tree Protocol (STP). Having the STP enabled might delay the FCoE Initialization Protocol (FIP) response at the switch and cause an all paths down (APD) condition.

  The FIP is a protocol that FCoE uses to discover and initialize FCoE entities on the Ethernet.

- Turn on Priority-based Flow Control (PFC) and set it to AUTO.

VMware recommends that you use the following firmware on the FCoE switch:

- Cisco Nexus 5000: version 4.1(3)N2 or higher.
- Brocade FCoE switch: version 6.3.1 or higher.

Network Adapter Best Practices

If you plan to enable software FCoE adapters to work with network adapters, specific considerations apply.

- Make sure that the latest microcode is installed on the FCoE network adapter.
- If the network adapter has multiple ports, when configuring networking, add each port to a separate vSwitch. This practice helps you to avoid an APD condition when a disruptive event, such as an MTU change, occurs.
- Do not move a network adapter port from one vSwitch to another when FCoE traffic is active. If you need to make this change, reboot your host afterwards.
- If you changed the vSwitch for a network adapter port and caused a failure, moving the port back to the original vSwitch resolves the problem.

Set Up Networking for Software FCoE

Before you activate the software FCoE adapters, you need to connect the VMkernel to physical FCoE NICs installed on your host.

Procedure

1 Log in to the vSphere Client, and select a host from the inventory panel.

2 Click the **Configuration** tab and click **Networking**.

3 In the vSphere standard switch view, click **Add Networking**.

4 Select **VMkernel** and click **Next**.

5 Select **Create a vSphere standard switch** to create a new vSphere standard switch.

6 Select the network adapter (vmnic#) that supports FCoE and click **Next**.

If your host has multiple network adapters or multiple ports on the adapter, you can add all of them to a single vSphere standard switch. An alternative is to connect each FCoE NIC to a separate standard switch.

NOTE ESXi 5.0 supports the maximum of four network adapter ports used for software FCoE.

7 Enter a network label.

Network label is a friendly name that identifies the VMkernel adapter that you are creating, for example, FCoE.

8 Specify a VLAN ID and click **Next**.

Because FCoE traffic requires an isolated network, make sure that the VLAN ID you enter is different from the one used for regular networking on your host. For more information, see the *vSphere Networking* documentation.

9 Specify the IP settings and click **Next**.

10 Review the information and click **Finish**.

You have created the virtual VMkernel adapter for the physical FCoE network adapter installed on your host.

NOTE To avoid FCoE traffic disruptions, do not remove the FCoE network adapter (vmnic#) from the vSphere standard switch after you set up FCoE networking.

Add Software FCoE Adapters

You must activate software FCoE adapters so that your host can use them to access Fibre Channel storage.

The number of software FCoE adapters you can activate corresponds to the number of physical FCoE NIC ports on your host. ESXi 5.0 supports the maximum of four software FCoE adapters on one host.

Prerequisites

Set up networking for the software FCoE adapter.

Required privilege: **Host.Configuration.Storage Partition Configuration**

Procedure

1 Log in to the vSphere Client, and select a host from the inventory panel.

2 Click the **Configuration** tab and click **Storage Adapters** in the Hardware panel.

3 Click **Add**, select **Software FCoE Adapter**, and click **OK**.

4 On the Add Software FCoE Adapter dialog box, select an appropriate vmnic from the drop-down list of physical network adapters.

   Only those adapters that are not yet used for FCoE traffic are listed.

5 Click **OK**.

   The software FCoE adapters appears on the list of storage adapters.

After you activate the software FCoE adapter, you can view its properties. If you do not use the adapter, you can remove it from the list of adapters.

N-Port ID Virtualization

N-Port ID Virtualization (NPIV) is an ANSI T11 standard that describes how a single Fibre Channel HBA port can register with the fabric using several worldwide port names (WWPNs). This allows a fabric-attached N-port to claim multiple fabric addresses. Each address appears as a unique entity on the Fibre Channel fabric.

How NPIV-Based LUN Access Works

NPIV enables a single FC HBA port to register several unique WWNs with the fabric, each of which can be assigned to an individual virtual machine.

SAN objects, such as switches, HBAs, storage devices, or virtual machines can be assigned World Wide Name (WWN) identifiers. WWNs uniquely identify such objects in the Fibre Channel fabric. When virtual machines have WWN assignments, they use them for all RDM traffic, so the LUNs pointed to by any of the RDMs on the virtual machine must not be masked against its WWNs. When virtual machines do not have WWN assignments, they access storage LUNs with the WWNs of their host's physical HBAs. By using NPIV, however, a SAN administrator can monitor and route storage access on a per virtual machine basis. The following section describes how this works.

When a virtual machine has a WWN assigned to it, the virtual machine's configuration file (.vmx) is updated to include a WWN pair (consisting of a World Wide Port Name, WWPN, and a World Wide Node Name, WWNN). As that virtual machine is powered on, the VMkernel instantiates a virtual port (VPORT) on the physical HBA which is used to access the LUN. The VPORT is a virtual HBA that appears to the FC fabric as a physical HBA, that is, it has its own unique identifier, the WWN pair that was assigned to the virtual machine. Each VPORT is specific to the virtual machine, and the VPORT is destroyed on the host and it no longer appears to the FC fabric when the virtual machine is powered off. When a virtual machine is migrated from one host to another, the VPORT is closed on the first host and opened on the destination host.

If NPIV is enabled, WWN pairs (WWPN & WWNN) are specified for each virtual machine at creation time. When a virtual machine using NPIV is powered on, it uses each of these WWN pairs in sequence to try to discover an access path to the storage. The number of VPORTs that are instantiated equals the number of physical HBAs present on the host. A VPORT is created on each physical HBA that a physical path is found on. Each physical path is used to determine the virtual path that will be used to access the LUN. Note that HBAs that are not NPIV-aware are skipped in this discovery process because VPORTs cannot be instantiated on them.

Requirements for Using NPIV

If you plan to enable NPIV on your virtual machines, you should be aware of certain requirements.

The following requirements exist:

- NPIV can be used only for virtual machines with RDM disks. Virtual machines with regular virtual disks use the WWNs of the host's physical HBAs.
- HBAs on your host must support NPIV.

  For information, see the *vSphere Compatibility Guide* and refer to you vendor documentation.

  - Use HBAs of the same type, either all QLogic or all Emulex. VMware does not support heterogeneous HBAs on the same host accessing the same LUNs.
  - If a host uses multiple physical HBAs as paths to the storage, zone all physical paths to the virtual machine. This is required to support multipathing even though only one path at a time will be active.
  - Make sure that physical HBAs on the host have access to all LUNs that are to be accessed by NPIV-enabled virtual machines running on that host.
- The switches in the fabric must be NPIV-aware.
- When configuring a LUN for NPIV access at the storage level, make sure that the NPIV LUN number and NPIV target ID match the physical LUN and Target ID.
- Use the vSphere Client to manipulate virtual machines with WWNs.

NPIV Capabilities and Limitations

Learn about specific capabilities and limitations of the use of NPIV with ESXi.

ESXi with NPIV supports the following items:

- NPIV supports vMotion. When you use vMotion to migrate a virtual machine it retains the assigned WWN.

  If you migrate an NPIV-enabled virtual machine to a host that does not support NPIV, VMkernel reverts to using a physical HBA to route the I/O.
- If your FC SAN environment supports concurrent I/O on the disks from an active-active array, the concurrent I/O to two different NPIV ports is also supported.

When you use ESXi with NPIV, the following limitations apply:

- Because the NPIV technology is an extension to the FC protocol, it requires an FC switch and does not work on the direct attached FC disks.
- When you clone a virtual machine or template with a WWN assigned to it, the clones do not retain the WWN.
- NPIV does not support Storage vMotion.
- Disabling and then re-enabling the NPIV capability on an FC switch while virtual machines are running can cause an FC link to fail and I/O to stop.

Assign WWNs to Virtual Machines

You can assign a WWN to a new virtual machine with an RDM disk when you create this virtual machine.

You can create from 1 to 16 WWN pairs, which can be mapped to the first 1 to 16 physical HBAs on the host.

Procedure

1 Open the New Virtual Machine wizard.

2 Select **Custom**, and click **Next**.

3 Follow all steps required to create a custom virtual machine.

4 On the Select a Disk page, select **Raw Device Mapping**, and click **Next**.

5 From a list of SAN disks or LUNs, select a raw LUN you want your virtual machine to access directly.

6 Select a datastore for the RDM mapping file.

You can place the RDM file on the same datastore where your virtual machine files reside, or select a different datastore.

NOTE If you want to use vMotion for a virtual machine with enabled NPIV, make sure that the RDM file is located on the same datastore where the virtual machine configuration file resides.

7 Follow the steps required to create a virtual machine with the RDM.

8 On the Ready to Complete page, select the **Edit the virtual machine settings before completion** check box and click **Continue**.

The Virtual Machine Properties dialog box opens.

9 Assign WWNs to the virtual machine.

a Click the **Options** tab, and select **Fibre Channel NPIV**.

b Select **Generate new WWNs**.

c Specify the number of WWNNs and WWPNs.

A minimum of 2 WWPNs are needed to support failover with NPIV. Typically only 1 WWNN is created for each virtual machine.

10 Click **Finish**.

The host creates WWN assignments for the virtual machine.

What to do next

Register newly created WWNs in the fabric so that the virtual machine is able to log in to the switch, and assign storage LUNs to the WWNs.

Modify WWN Assignments

You can modify WWN assignments for a virtual machine with an RDM.

Typically, you do not need to change existing WWN assignments on your virtual machine. In certain circumstances, for example, when manually assigned WWNs are causing conflicts on the SAN, you might need to change or remove WWNs.

Prerequisites

Make sure to power off the virtual machine if you want to edit the existing WWNs.

Before you begin, ensure that your SAN administrator has provisioned the storage LUN ACL to allow the virtual machine's ESXi host to access it.

Procedure

1 Open the Virtual Machine Properties dialog box by clicking the **Edit Settings** link for the selected virtual machine.

2 Click the **Options** tab and select **Fibre Channel NPIV**.

The Virtual Machine Properties dialog box opens.

3 Edit the WWN assignments by selecting one of the following options:

Option	Description
Temporarily disable NPIV for this virtual machine	Disable the WWN assignments for the virtual machine.
Leave unchanged	The existing WWN assignments are retained. The read-only WWN Assignments section of this dialog box displays the node and port values of any existing WWN assignments.
Generate new WWNs	New WWNs are generated and assigned to the virtual machine, overwriting any existing WWNs (those of the HBA itself are unaffected).
Remove WWN assignment	The WWNs assigned to the virtual machine are removed and it uses the HBA WWNs to access the storage LUN. This option is not available if you are creating a new virtual machine.

4 Click **OK** to save your changes.

Modifying Fibre Channel Storage for ESXi 5

This section discusses many of the storage devices supported in conjunction with VMware ESXi. For each device, it lists the major known potential issues, points to vendor-specific information (if available), and includes information from VMware knowledge base articles.

NOTE Information related to specific storage devices is updated only with each release. New information might already be available. Consult the most recent Storage/SAN Compatibility, check with your storage array vendor, and explore the VMware knowledge base articles.

This chapter includes the following topics:

- "Testing ESXi SAN Configurations," on page 43
- "General Setup Considerations for Fibre Channel SAN Arrays," on page 44
- "EMC CLARiiON Storage Systems," on page 44
- "EMC Symmetrix Storage Systems," on page 45
- "IBM System Storage DS4800 Storage Systems," on page 46
- "IBM Systems Storage 8000 and IBM ESS800," on page 47
- "HP StorageWorks Storage Systems," on page 47
- "Hitachi Data Systems Storage," on page 48
- "Network Appliance Storage," on page 48
- "LSI-Based Storage Systems," on page 49

Testing ESXi SAN Configurations

ESXi supports a variety of SAN storage systems in different configurations. Generally, VMware tests ESXi with supported storage systems for basic connectivity, HBA failover, and so on.

Not all storage devices are certified for all features and capabilities of ESXi, and vendors might have specific positions of support with regard to ESXi.

Basic connectivity Tests whether ESXi can recognize and operate with the storage array. This configuration does not allow for multipathing or any type of failover.

HBA failover The server is equipped with multiple HBAs connecting to one or more SAN switches. The server is robust to HBA and switch failure only.

Storage port failover The server is attached to multiple storage ports and is robust to storage port failures and switch failures.

Boot from SAN	The host boots from a LUN configured on the SAN rather than from the server itself.
Direct connect	The server connects to the array without using switches. For all other tests, a fabric connection is used. FC Arbitrated Loop (AL) is not supported.
Clustering	The system is tested with Microsoft Cluster Service running in the virtual machine.

General Setup Considerations for Fibre Channel SAN Arrays

When you prepare your FC SAN storage to work with ESXi, you must follow specific general requirements that apply to all storage arrays.

For all storage arrays, make sure that the following requirements are met:

- LUNs must be presented to each HBA of each host with the same LUN ID number.

  Because instructions on how to configure identical SAN LUN IDs are vendor specific, consult your storage array documentation for more information.

- Unless specified for individual storage arrays, set the host type for LUNs presented to ESXi to `Linux`, `Linux Cluster`, or, if available, to `vmware` or `esx`.

- If you are using vMotion, DRS, or HA, make sure that both source and target hosts for virtual machines can see the same LUNs with identical LUN IDs.

  SAN administrators might find it counterintuitive to have multiple hosts see the same LUNs because they might be concerned about data corruption. However, VMFS prevents multiple virtual machines from writing to the same file at the same time, so provisioning the LUNs to all required ESXi system is appropriate.

EMC CLARiiON Storage Systems

EMC CLARiiON storage systems work with ESXi hosts in SAN configurations.

Basic configuration includes the following steps:

1 Installing and configuring the storage device.

2 Configuring zoning at the switch level.

3 Creating RAID groups.

4 Creating and binding LUNs.

5 Registering the servers connected to the SAN. By default, the host automatically performs this step.

6 Creating storage groups that contain the servers and LUNs.

Use the EMC storage management software to perform configuration. For information, see the EMC documentation.

ESXi automatically sends the host's name and IP address to the array and registers the host with the array. You are no longer required to perform host registration manually. However, if you prefer to use storage management software, such as EMC Navisphere, to perform manual registration, turn off the ESXi auto-registration. Turning it off helps you avoid overwriting the manual user registration. For information, see "Disable Automatic Host Registration," on page 58.

Because this array is an active-passive disk array, the following general considerations apply.

- The default multipathing policy for CLARiiON arrays that do not support ALUA is Most Recently Used. For CLARiiON arrays that support ALUA, the default multipathing policy is VMW_PSP_FIXED. The ESXi system sets the default policy when it identifies the array.

- Automatic volume resignaturing is not supported for AX100 storage devices.
- To use boot from SAN, make sure that the active SP is chosen for the boot LUN's target in the HBA BIOS.

IMPORTANT For ESXi to support EMC CLARiiON with ALUA, check the HCLs to make sure that you use the correct firmware version on the storage array. For additional information, contact your storage vendor.

EMC CLARiiON AX100 and RDM

On EMC CLARiiON AX100 systems, RDMs are supported only if you use the Navisphere Management Suite for SAN administration. Navilight is not guaranteed to work properly.

To use RDMs successfully, a given LUN must be presented with the same LUN ID to every ESXi host in the cluster. By default, the AX100 does not support this configuration.

EMC CLARiiON AX100 Display Problems with Inactive Connections

When you use an AX100 FC storage device directly connected to an ESXi system, you must verify that all connections are operational and unregister any connections that are no longer in use. If you do not, ESXi cannot discover new LUNs or paths.

Consider the following scenario:

An ESXi host is directly connected to an AX100 storage device. The host has two FC HBAs. One of the HBAs was previously registered with the storage array and its LUNs were configured, but the connections are now inactive.

When you connect the second HBA on the host to the AX100 and register it, the host correctly shows the array as having an active connection. However, none of the LUNs that were previously configured to the host are visible, even after repeated rescans.

To resolve this issue, remove the inactive HBA, unregister the connection to the inactive HBA, or make all inactive connections active. This causes only active HBAs to be in the storage group. After this change, rescan to add the configured LUNs.

EMC Symmetrix Storage Systems

EMC Symmetrix storage systems work with ESXi hosts in FC SAN configurations. Generally, you use the EMC software to perform configurations.

The following settings are required on the Symmetrix networked storage system. For more information, see the EMC documentation.

- Common serial number (C)
- Auto negotiation (EAN) enabled
- Fibrepath enabled on this port (VCM)
- SCSI 3 (SC3) set enabled
- Unique world wide name (UWN)
- SPC-2 (Decal) (SPC2) SPC-2 flag is required

The ESXi host considers any LUNs from a Symmetrix storage array with a capacity of 50MB or less as management LUNs. These LUNs are also known as pseudo or gatekeeper LUNs. These LUNs appear in the EMC Symmetrix Management Interface and should not be used to hold data.

IBM System Storage DS4800 Storage Systems

IBM System Storage DS4800 systems used to be called IBM FAStT. A number of storage array vendors (including LSI and StorageTek) make SAN storage arrays that are compatible with the DS4800.

For your host to work with IBM DS4800 and other compatible systems, make sure to set the multipathing policy on your host to Most Recently Used.

Configuring the Hardware for SAN Failover with DS4800 Storage Servers

This topic provides information on how to set up a highly available SAN failover configuration with an ESXi host and DS4800 storage.

You must have the following hardware components:

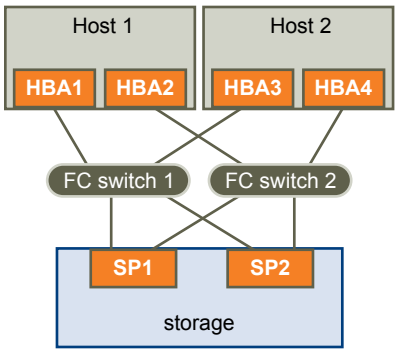
- Two FC HBAs, such as QLogic or Emulex, on each ESXi machine.
- Two FC switches connecting the HBAs to the SAN (for example, FC switch 1 and FC switch 2).
- Two SPs (for example, SP1 and SP2).

  Each SP must have at least two ports connected to the SAN.

Use the following connection settings for the ESXi host, as shown in SAN Failover:

- Connect each HBA on each ESXi machine to a separate switch. For example, connect HBA1 to FC switch 1 and HBA2 to FC switch 2.

Figure 5-1. SAN Failover



This configuration provides two paths from each HBA, so that each element of the connection can fail over to a redundant path. The order of the paths in this configuration provides HBA and switch failover without the need to trigger SP failover. The storage processor that the preferred paths are connected to must own the LUNs. In the preceding example configuration, SP1 owns them.

NOTE The preceding example assumes that the switches are not connected through an Inter-Switch Link (ISL) in one fabric.

Disabling Auto Volume Transfer

To avoid the possibility of path thrashing, disable Auto Volume Transfer (AVT) on the SAN storage processors. If AVT is enabled, the two storage processors can alternately take ownership of the LUN in certain situations, resulting in performance degradation. AVT is also known as ADT (Auto Disk Transfer).

To disable AVT, in the DS 4800 Storage Manager, for each port defined in each host group that contains HBAs for one or more ESXi hosts, set the host type to `LNXCL` or, in later versions, to `VMware`.

You must reboot the hosts after you change the AVT configuration.

Configure Storage Processor Sense Data

A DS4800 SP that runs Windows as a guest operating system should return `Not Ready` sense data when it is quiescent. Returning `Unit Attention` might cause the Windows guest to fail during a failover.

Procedure

1 Determine the index for the `LNXCL` host type by using the following commands in a shell window.

Press Enter after each command.

`SMcli.exe` *`ip-addr-for-SPA`* `show hosttopology; Enter SMcli.exe` *`ip-addr-for-SPB`* `show hosttopology`

The following commands assume that `13` is the index corresponding to `LNXCL` in the NVSRAM host type definitions. If your storage processors have `LNXCL` at a different index, substitute that index for `13` in the following commands.

2 Execute these commands for SPA to have it return `Not Ready` sense data.

Press Enter only after you enter all commands.**`SMcli.exe` *`ip-addr-for-SPA`* `set controller [a] HostNVSRAMBYTE [13,0x12]=0x01; set controller [a] HostNVSRAMBYTE [13,0x13]=0x00; reset Controller [a]`**

3 Execute these commands for SPB to have it return `Not Ready` sense data.

Press Enter only after you enter all commands.**`SMcli.exe` *`ip-addr-for-SPB`* `set controller [b] HostNVSRAMBYTE [13,0x12]=0x01; set controller [b] HostNVSRAMBYTE [13,0x13]=0x00; reset Controller [b]`**

NOTE If you use the DS4800 Storage Manager GUI, paste the configuration commands for both storage processors into a single script and configure both storage processors at the same time. If you use `SMcli.exe`, make individual connections to each SP.

IBM Systems Storage 8000 and IBM ESS800

The IBM Systems Storage 8000 and IBM ESS800 systems use an active-active array that does not need special configuration in conjunction with VMware ESXi.

The following considerations apply when you use these systems:

- Automatic resignaturing is not supported for these systems.
- To use RDMs successfully, a given LUN must be presented with the same LUN ID to every ESXi host in the cluster.
- In the ESS800 Configuration Management tool, select **Use same ID for LUN in source and target**.
- If you are configuring the host to use boot from SAN from these arrays, disable the internal fibre port for the corresponding blade until installation is finished.

HP StorageWorks Storage Systems

This section includes configuration information for the different HP StorageWorks storage systems.

For additional information, see the HP ActiveAnswers section on VMware ESXi at the HP web site.

HP StorageWorks EVA

To use an HP StorageWorks EVA system with ESXi, you must configure the correct host mode type.

Set the connection type to Custom when you present a LUN to the host. The value is one of the following:

- For EVA4000/6000/8000 active-active arrays with firmware below 5.031, use the host mode type `000000202200083E`.
- For EVA4000/6000/8000 active-active arrays with firmware 5.031 and above, use the host mode type `VMware`.

Otherwise, EVA systems do not require special configuration changes to work with an ESXi system.

See the VMware Infrastructure, HP StorageWorks Best Practices at the HP Web site.

HP StorageWorks XP

For HP StorageWorks XP, you need to set the host mode to specific parameters.

- On XP128/1024/10000/12000, set the host mode to `Windows (0x0C)`.
- On XP24000/20000, set the host mode to `0x01`.

Hitachi Data Systems Storage

This section introduces the setup for Hitachi Data Systems storage. This storage solution is also available from Sun and as HP XP storage.

LUN masking — To mask LUNs on an ESXi host, use the HDS Storage Navigator software for best results.

Microcode and configurations — Check with your HDS representative for exact configurations and microcode levels needed for interoperability with ESXi. If your microcode is not supported, interaction with ESXi is usually not possible.

Modes — The modes you set depend on the model you are using, for example:

- 9900 and 9900v uses Netware host mode.
- 9500v series uses Hostmode1: standard and Hostmode2: SUN Cluster.

Check with your HDS representative for host mode settings for the models not listed here.

Network Appliance Storage

When configuring a Network Appliance storage device, first set the appropriate LUN type and initiator group type for the storage array.

LUN type — VMware (if VMware type is not available, use Linux).

Initiator group type — VMware (if VMware type is not available, use Linux).

You must then provision storage.

LSI-Based Storage Systems

During ESXi installation, do not present the management LUN, also known as access LUN, from the LSI-based arrays to the host.

Otherwise, ESXi installation might fail.

Booting ESXi from Fibre Channel SAN 6

When you set up your host to boot from a SAN, your host's boot image is stored on one or more LUNs in the SAN storage system. When the host starts, it boots from the LUN on the SAN rather than from its local disk.

ESXi supports booting through a Fibre Channel host bus adapter (HBA) or a Fibre Channel over Ethernet (FCoE) converged network adapter (CNA).

This chapter includes the following topics:

- "Boot from SAN Benefits," on page 51
- "Boot from Fibre Channel SAN Requirements and Considerations," on page 52
- "Getting Ready for Boot from SAN," on page 52
- "Configure Emulex HBA to Boot from SAN," on page 53
- "Configure QLogic HBA to Boot from SAN," on page 55

Boot from SAN Benefits

Boot from SAN can provide numerous benefits to your environment. However, in certain cases, you should not use boot from SAN for ESXi hosts. Before you set up your system for boot from SAN, decide whether it is appropriate for your environment.

CAUTION When you use boot from SAN with multiple ESXi hosts, each host must have its own boot LUN. If you configure multiple hosts to share the same boot LUN, ESXi image corruption is likely to occur.

If you use boot from SAN, the benefits for your environment will include the following:

- Cheaper servers. Servers can be more dense and run cooler without internal storage.
- Easier server replacement. You can replace servers and have the new server point to the old boot location.
- Less wasted space. Servers without local disks often take up less space.
- Easier backup processes. You can backup the system boot images in the SAN as part of the overall SAN backup procedures. Also, you can use advanced array features such as snapshots on the boot image.
- Improved management. Creating and managing the operating system image is easier and more efficient.
- Better reliability. You can access the boot disk through multiple paths, which protects the disk from being a single point of failure.

Boot from Fibre Channel SAN Requirements and Considerations

Your ESXi boot configuration must meet specific requirements.

Table 6-1. Boot from SAN Requirements

Requirement	Description
ESXi system requirements	Follow vendor recommendation for the server booting from a SAN.
Adapter requirements	Enable and correctly configure the adapter, so it can access the boot LUN. See your vendor documentation.
Access control	■ Each host must have access to its own boot LUN only, not the boot LUNs of other hosts. Use storage system software to make sure that the host accesses only the designated LUNs. ■ Multiple servers can share a diagnostic partition. You can use array specific LUN masking to achieve this.
Multipathing support	Multipathing to a boot LUN on active-passive arrays is not supported because the BIOS does not support multipathing and is unable to activate a standby path.
SAN considerations	SAN connections must be through a switched topology if the array is not certified for direct connect topology. If the array is certified for direct connect topology, the SAN connections can be made directly to the array. Boot from SAN is supported for both switched topology and direct connect topology if these topologies for the specific array are certified.
Hardware- specific considerations	If you are running an IBM eServer BladeCenter and use boot from SAN, you must disable IDE drives on the blades.

Getting Ready for Boot from SAN

When you set up your boot from SAN environment, you perform a number of tasks.

This section describes the generic boot-from-SAN enablement process on the rack mounted servers. For information on enabling boot from SAN on Cisco Unified Computing System FCoE blade servers, refer to Cisco documentation.

1 Configure SAN Components and Storage System on page 52

Before you set up your ESXi host to boot from a SAN LUN, configure SAN components and a storage system.

2 Configure Storage Adapter to Boot from SAN on page 53

When you set up your host to boot from SAN, you enable the boot adapter in the host BIOS. You then configure the boot adapter to initiate a primitive connection to the target boot LUN.

3 Set Up Your System to Boot from Installation Media on page 53

When setting up your host to boot from SAN, you first boot the host from the VMware installation media. To achieve this, you need to change the system boot sequence in the BIOS setup.

Configure SAN Components and Storage System

Before you set up your ESXi host to boot from a SAN LUN, configure SAN components and a storage system.

Because configuring the SAN components is vendor specific, refer to the product documentation for each item.

Procedure

1 Connect network cable, referring to any cabling guide that applies to your setup.

Check the switch wiring, if there is any.

2 Configure the storage array.

   a From the SAN storage array, make the ESXi host visible to the SAN. This process is often referred to as creating an object.

   b From the SAN storage array, set up the host to have the WWPNs of the host's adapters as port names or node names.

   c Create LUNs.

   d Assign LUNs.

   e Record the IP addresses of the switches and storage arrays.

   f Record the WWPN for each SP.

CAUTION If you use scripted installation to install ESXi in boot from SAN mode, you need to take special steps to avoid unintended data loss.

Configure Storage Adapter to Boot from SAN

When you set up your host to boot from SAN, you enable the boot adapter in the host BIOS. You then configure the boot adapter to initiate a primitive connection to the target boot LUN.

Prerequisites

Determine the WWPN for the storage adapter.

Procedure

- Configure the storage adapter to boot from SAN.

  Because configuring boot adapters is vendor specific, refer to your vendor documentation.

Set Up Your System to Boot from Installation Media

When setting up your host to boot from SAN, you first boot the host from the VMware installation media. To achieve this, you need to change the system boot sequence in the BIOS setup.

Because changing the boot sequence in the BIOS is vendor specific, refer to vendor documentation for instructions. The following procedure explains how to change the boot sequence on an IBM host.

Procedure

1 During your system power up, enter the system BIOS Configuration/Setup Utility.

2 Select **Startup Options** and press Enter.

3 Select **Startup Sequence Options** and press Enter.

4 Change the **First Startup Device** to **[CD-ROM]**.

You can now install ESXi.

Configure Emulex HBA to Boot from SAN

Configuring the Emulex HBA BIOS to boot from SAN includes enabling the BootBIOS prompt and enabling BIOS.

Procedure

1 Enable the BootBIOS Prompt on page 54

  When you configure the Emulex HBA BIOS to boot ESXi from SAN, you need to enable the BootBIOS prompt.

2 Enable the BIOS on page 54
When you configure the Emulex HBA BIOS to boot ESXi from SAN, you need to enable BIOS.

Enable the BootBIOS Prompt

When you configure the Emulex HBA BIOS to boot ESXi from SAN, you need to enable the BootBIOS prompt.

Procedure

1 Run **`lputil`**.

2 Select **3. Firmware Maintenance**.

3 Select an adapter.

4 Select **6. Boot BIOS Maintenance**.

5 Select **1. Enable Boot BIOS**.

Enable the BIOS

When you configure the Emulex HBA BIOS to boot ESXi from SAN, you need to enable BIOS.

Procedure

1 Reboot the host.

2 To configure the adapter parameters, press ALT+E at the Emulex prompt and follow these steps.

a Select an adapter (with BIOS support).

b Select **2. Configure This Adapter's Parameters.**

c Select **1. Enable or Disable BIOS.**

d Select **1** to enable BIOS.

e Select **x** to exit and **Esc** to return to the previous menu.

3 To configure the boot device, follow these steps from the Emulex main menu.

a Select the same adapter.

b Select **1. Configure Boot Devices.**

c Select the location for the Boot Entry.

d Enter the two-digit boot device.

e Enter the two-digit (HEX) starting LUN (for example, **`08`**).

f Select the boot LUN.

g Select **1. WWPN.** (Boot this device using WWPN, not DID).

h Select **x** to exit and **Y** to reboot.

4 Boot into the system BIOS and move Emulex first in the boot controller sequence.

5 Reboot and install on a SAN LUN.

Configure QLogic HBA to Boot from SAN

This sample procedure explains how to configure the QLogic HBA to boot ESXi from SAN. The procedure involves enabling the QLogic HBA BIOS, enabling the selectable boot, and selecting the boot LUN.

Procedure

1 While booting the server, press **Ctrl+Q** to enter the Fast!UTIL configuration utility.

2 Perform the appropriate action depending on the number of HBAs.

Option	Description
One HBA	If you have only one host bus adapter (HBA), the Fast!UTIL Options page appears. Skip to Step 3.
Multiple HBAs	If you have more than one HBA, select the HBA manually. a In the Select Host Adapter page, use the arrow keys to position the cursor on the appropriate HBA. b Press **Enter**.

3 In the Fast!UTIL Options page, select **Configuration Settings** and press **Enter**.

4 In the Configuration Settings page, select **Adapter Settings** and press **Enter**.

5 Set the BIOS to search for SCSI devices.

   a In the Host Adapter Settings page, select **Host Adapter BIOS**.

   b Press **Enter** to toggle the value to Enabled.

   c Press **Esc** to exit.

6 Enable the selectable boot.

   a Select **Selectable Boot Settings** and press **Enter**.

   b In the Selectable Boot Settings page, select **Selectable Boot**.

   c Press **Enter** to toggle the value to **Enabled**.

7 Use the cursor keys to select the Boot Port Name entry in the list of storage processors (SPs) and press **Enter** to open the Select Fibre Channel Device screen.

8 Use the cursor keys to select the specific SP and press **Enter**.

   If you are using an active-passive storage array, the selected SP must be on the preferred (active) path to the boot LUN. If you are not sure which SP is on the active path, use your storage array management software to find out. The target IDs are created by the BIOS and might change with each reboot.

9 Perform the appropriate action depending on the number of LUNs attached to the SP.

Option	Description
One LUN	The LUN is selected as the boot LUN. You do not need to enter the Select LUN screen.
Multiple LUNs	Select LUN screen opens. Use the cursor to select the boot LUN, then press **Enter**.

10 If any remaining storage processors show in the list, press **C** to clear the data.

11 Press **Esc** twice to exit and press **Enter** to save the setting.

Best Practices for Fibre Channel Storage 7

When using ESXi with Fibre Channel SAN, follow best practices that VMware offers to avoid performance problems.

The vSphere Client offers extensive facilities for collecting performance information. The information is graphically displayed in the vSphere Client. The vSphere Client updates its display periodically.

You can also use the `resxtop` or `esxtop` command-line utilities. The utilities provide a detailed look at how ESXi uses resources in real time. For more information, see the *vSphere Resource Management* documentation.

Check with your storage representative if your storage system supports Storage API - Array Integration hardware acceleration features. If it does, refer to your vendor documentation for information on how to enable hardware acceleration support on the storage system side. For more information, see Chapter 18, "Storage Hardware Acceleration," on page 173.

This chapter includes the following topics:

- "Preventing Fibre Channel SAN Problems," on page 57
- "Disable Automatic Host Registration," on page 58
- "Optimizing Fibre Channel SAN Storage Performance," on page 58
- "Fibre Channel SAN Configuration Checklist," on page 59

Preventing Fibre Channel SAN Problems

When using ESXi in conjunction with a Fibre Channel SAN, you must follow specific guidelines to avoid SAN problems.

You should observe these tips for preventing problems with your SAN configuration:

- Place only one VMFS datastore on each LUN.
- Do not change the path policy the system sets for you unless you understand the implications of making such a change.
- Document everything. Include information about zoning, access control, storage, switch, server and FC HBA configuration, software and firmware versions, and storage cable plan.
- Plan for failure:
  - Make several copies of your topology maps. For each element, consider what happens to your SAN if the element fails.
  - Cross off different links, switches, HBAs and other elements to ensure you did not miss a critical failure point in your design.

- Ensure that the Fibre Channel HBAs are installed in the correct slots in the host, based on slot and bus speed. Balance PCI bus load among the available busses in the server.
- Become familiar with the various monitor points in your storage network, at all visibility points, including host's performance charts, FC switch statistics, and storage performance statistics.
- Be cautious when changing IDs of the LUNs that have VMFS datastores being used by your ESXi host. If you change the ID, the datastore becomes inactive and its virtual machines fail. You can resignature the datastore to make it active again. See "Managing Duplicate VMFS Datastores," on page 122.

  If there are no running virtual machines on the VMFS datastore, after you change the ID of the LUN, you must use rescan to reset the ID on your host. For information on using rescan, see "Perform Storage Rescan," on page 124.

Disable Automatic Host Registration

When you use EMC CLARiiON or Invista arrays for storage, it is required that the hosts register with the arrays. ESXi performs automatic host registration by sending the host's name and IP address to the array. If you prefer to perform manual registration using storage management software, disable the ESXi auto-registration feature.

Procedure

1 In the vSphere Client, select the host in the inventory panel.

2 Click the **Configuration** tab and click **Advanced Settings** under Software.

3 Click **Disk** in the left panel and scroll down to Disk.EnableNaviReg on the right.

4 Change the default value to 0.

This disables the automatic host registration enabled by default.

Optimizing Fibre Channel SAN Storage Performance

Several factors contribute to optimizing a typical SAN environment.

If the environment is properly configured, the SAN fabric components (particularly the SAN switches) are only minor contributors because of their low latencies relative to servers and storage arrays. Make sure that the paths through the switch fabric are not saturated, that is, that the switch fabric is running at the highest throughput.

Storage Array Performance

Storage array performance is one of the major factors contributing to the performance of the entire SAN environment.

If there are issues with storage array performance, be sure to consult your storage array vendor's documentation for any relevant information.

Follow these general guidelines to improve the array performance in the vSphere environment:

- When assigning LUNs, remember that each LUN is accessed by a number of hosts, and that a number of virtual machines can run on each host. One LUN used by a host can service I/O from many different applications running on different operating systems. Because of this diverse workload, the RAID group containing the ESXi LUNs should not include LUNs used by other servers that are not running ESXi.
- Make sure read/write caching is enabled.

- SAN storage arrays require continual redesign and tuning to ensure that I/O is load balanced across all storage array paths. To meet this requirement, distribute the paths to the LUNs among all the SPs to provide optimal load balancing. Close monitoring indicates when it is necessary to rebalance the LUN distribution.

  Tuning statically balanced storage arrays is a matter of monitoring the specific performance statistics (such as I/O operations per second, blocks per second, and response time) and distributing the LUN workload to spread the workload across all the SPs.

  NOTE Dynamic load balancing is not currently supported with ESXi.

Server Performance with Fibre Channel

You must consider several factors to ensure optimal server performance.

Each server application must have access to its designated storage with the following conditions:

- High I/O rate (number of I/O operations per second)
- High throughput (megabytes per second)
- Minimal latency (response times)

Because each application has different requirements, you can meet these goals by choosing an appropriate RAID group on the storage array. To achieve performance goals:

- Place each LUN on a RAID group that provides the necessary performance levels. Pay attention to the activities and resource utilization of other LUNS in the assigned RAID group. A high-performance RAID group that has too many applications doing I/O to it might not meet performance goals required by an application running on the ESXi host.
- Make sure that each server has a sufficient number of HBAs to allow maximum throughput for all the applications hosted on the server for the peak period. I/O spread across multiple HBAs provide higher throughput and less latency for each application.
- To provide redundancy in the event of HBA failure, make sure the server is connected to a dual redundant fabric.
- When allocating LUNs or RAID groups for ESXi systems, multiple operating systems use and share that resource. As a result, the performance required from each LUN in the storage subsystem can be much higher if you are working with ESXi systems than if you are using physical machines. For example, if you expect to run four I/O intensive applications, allocate four times the performance capacity for the ESXi LUNs.
- When using multiple ESXi systems in conjunction with vCenter Server, the performance needed from the storage subsystem increases correspondingly.
- The number of outstanding I/Os needed by applications running on an ESXi system should match the number of I/Os the HBA and storage array can handle.

Fibre Channel SAN Configuration Checklist

This topic provides a checklist of special setup requirements for different storage arrays and ESXi hosts.

Table 7-1. Multipathing Setup Requirements

Component	Comments
All storage arrays	Write cache must be disabled if not battery backed.
Topology	No single failure should cause both HBA and SP failover, especially with active-passive storage arrays.

Table 7-1. Multipathing Setup Requirements (Continued)

Component	Comments
IBM TotalStorage DS 4000 (formerly FastT)	Host type must be `LNXCL or VMware` in later versions. AVT (Auto Volume Transfer) is disabled in this host mode.
HDS 99xx and 95xxV family	HDS 9500V family (Thunder) requires two host modes: ■ Host Mode 1: `Standard`. ■ Host Mode 2: `Sun Cluster` HDS 99xx family (Lightning) and HDS Tabma (USP) require host mode set to `Netware`.
EMC Symmetrix	Enable the SPC2 and SC3 settings. Contact EMC for the latest settings.
EMC Clariion	Set the EMC Clariion failover mode to 1 or 4. Contact EMC for details.
HP MSA	Host type must be `Linux`. Set the connection type for each HBA port to `Linux`.
HP EVA	For EVA4000/6000/8000 firmware 5.031 and above, set the host type to `VMware`. Otherwise, set the host mode type to `Custom`. The value is: 000000202200083E.
HP XP	■ On XP128/1024/10000/12000, set the host mode to `Windows (0x0C)`. ■ On XP24000/20000, set the host mode to `0x01`.
NetApp	No specific requirements

ESXi Configuration	■ For all LUNs hosting clustered disks on active-passive arrays, use the Most Recently Used PSP.
	■ For LUNs on active-active arrays, you can use the Most Recently Used or Fixed PSP.
	■ With either active-passive or active-active arrays, you can use the Round Robin PSP.
	■ All FC HBAs must be of the same model.
	■ Set the following Software Advanced Settings for the host:
	  ■ Set Disk.UseLunReset to 1
	  ■ Set Disk.UseDeviceReset to 0

Using ESXi with iSCSI SAN 8

You can use ESXi in conjunction with a storage area network (SAN), a specialized high-speed network that connects computer systems to high-performance storage subsystems. Using ESXi together with a SAN provides storage consolidation, improves reliability, and helps with disaster recovery.

To use ESXi effectively with a SAN, you must have a working knowledge of ESXi systems and SAN concepts. Also, when you set up ESXi hosts to use Internet SCSI (iSCSI) SAN storage systems, you must be aware of certain special considerations that exist.

This chapter includes the following topics:

- "iSCSI SAN Concepts," on page 61
- "How Virtual Machines Access Data on an iSCSI SAN," on page 66

iSCSI SAN Concepts

If you are an administrator who plans to set up ESXi hosts to work with iSCSI SANs, you must have a working knowledge of iSCSI concepts.

iSCSI SANs use Ethernet connections between computer systems, or host servers, and high performance storage subsystems. The SAN components include iSCSI host bus adapters (HBAs) or Network Interface Cards (NICs) in the host servers, switches and routers that transport the storage traffic, cables, storage processors (SPs), and storage disk systems.

iSCSI SAN uses a client-server architecture. The client, called iSCSI initiator, operates on your host. It initiates iSCSI sessions by issuing SCSI commands and transmitting them, encapsulated into iSCSI protocol, to a server. The server is known as an iSCSI target. The iSCSI target represents a physical storage system on the network. It can also be provided by a virtual iSCSI SAN, for example, an iSCSI target emulator running in a virtual machine. The iSCSI target responds to the initiator's commands by transmitting required iSCSI data.

iSCSI Multipathing

When transferring data between the host server and storage, the SAN uses a technique known as multipathing. Multipathing allows you to have more than one physical path from the ESXi host to a LUN on a storage system.

Generally, a single path from a host to a LUN consists of an iSCSI adapter or NIC, switch ports, connecting cables, and the storage controller port. If any component of the path fails, the host selects another available path for I/O. The process of detecting a failed path and switching to another is called path failover.

For more information on multipathing, see Chapter 17, "Understanding Multipathing and Failover," on page 153.

Ports in the iSCSI SAN

A single discoverable entity on the iSCSI SAN, such as an initiator or a target, represents an iSCSI node. Each node has one or more ports that connect it to the SAN.

iSCSI ports are end-points of an iSCSI session. Each node can be identified in a number of ways.

IP Address Each iSCSI node can have an IP address associated with it so that routing and switching equipment on your network can establish the connection between the server and storage. This address is just like the IP address that you assign to your computer to get access to your company's network or the Internet.

iSCSI Name A worldwide unique name for identifying the node. iSCSI uses the iSCSI Qualified Name (IQN) and Extended Unique Identifier (EUI).

By default, ESXi generates unique iSCSI names for your iSCSI initiators, for example, `iqn.1998-01.com.vmware:iscsitestox-68158ef2`. Usually, you do not have to change the default value, but if you do, make sure that the new iSCSI name you enter is worldwide unique.

iSCSI Alias A more manageable name for an iSCSI device or port used instead of the iSCSI name. iSCSI aliases are not unique and are intended to be just a friendly name to associate with a port.

iSCSI Naming Conventions

iSCSI uses a special unique name to identify an iSCSI node, either target or initiator. This name is similar to the WorldWide Name (WWN) associated with Fibre Channel devices and is used as a way to universally identify the node.

iSCSI names are formatted in two different ways. The most common is the IQN format.

For more details on iSCSI naming requirements and string profiles, see RFC 3721 and RFC 3722 on the IETF Web site.

iSCSI Qualified Name (IQN) Format

The IQN format takes the form `iqn.yyyy-mm.naming-authority:unique name`, where:

- *yyyy-mm* is the year and month when the naming authority was established.
- *naming-authority* is usually reverse syntax of the Internet domain name of the naming authority. For example, the iscsi.vmware.com naming authority could have the iSCSI qualified name form of iqn. 1998-01.com.vmware.iscsi. The name indicates that the vmware.com domain name was registered in January of 1998, and iscsi is a subdomain, maintained by vmware.com.
- *unique name* is any name you want to use, for example, the name of your host. The naming authority must make sure that any names assigned following the colon are unique, such as:
  - iqn.1998-01.com.vmware.iscsi:name1
  - iqn.1998-01.com.vmware.iscsi:name2
  - iqn.1998-01.com.vmware.iscsi:name999

Enterprise Unique Identifier (EUI) Format

The EUI format takes the form `eui.16 hex digits`.

For example, `eui.0123456789ABCDEF`.

The 16-hexadecimal digits are text representations of a 64-bit number of an IEEE EUI (extended unique identifier) format. The top 24 bits are a company ID that IEEE registers with a particular company. The lower 40 bits are assigned by the entity holding that company ID and must be unique.

iSCSI Initiators

To access iSCSI targets, your host uses iSCSI initiators. The initiators transport SCSI requests and responses, encapsulated into the iSCSI protocol, between the host and the iSCSI target.

Your host supports different types of initiators.

For information on configuring and using iSCSI adapters, see Chapter 9, "Configuring iSCSI Adapters and Storage," on page 67.

Software iSCSI Adapter

A software iSCSI adapter is a VMware code built into the VMkernel. It allows your host to connect to the iSCSI storage device through standard network adapters. The software iSCSI adapter handles iSCSI processing while communicating with the network adapter. With the software iSCSI adapter, you can use iSCSI technology without purchasing specialized hardware.

Hardware iSCSI Adapter

A hardware iSCSI adapter is a third-party adapter that offloads iSCSI and network processing from your host. Hardware iSCSI adapters are divided into categories.

Dependent Hardware iSCSI Adapter

Depends on VMware networking, and iSCSI configuration and management interfaces provided by VMware.

This type of adapter can be a card that presents a standard network adapter and iSCSI offload functionality for the same port. The iSCSI offload functionality depends on the host's network configuration to obtain the IP, MAC, and other parameters used for iSCSI sessions. An example of a dependent adapter is the iSCSI licensed Broadcom 5709 NIC.

Independent Hardware iSCSI Adapter

Implements its own networking and iSCSI configuration and management interfaces.

An example of an independent hardware iSCSI adapter is a card that either presents only iSCSI offload functionality or iSCSI offload functionality and standard NIC functionality. The iSCSI offload functionality has independent configuration management that assigns the IP, MAC, and other parameters used for the iSCSI sessions. An example of a independent adapter is the QLogic QLA4052 adapter.

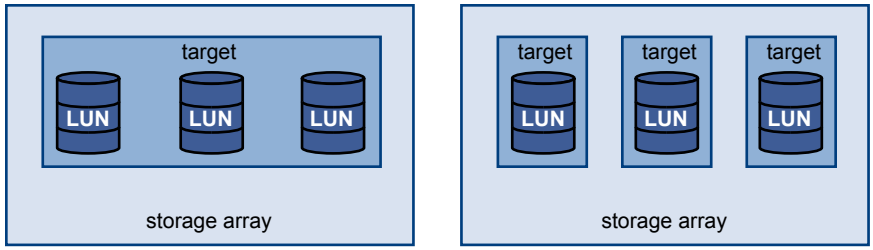
Hardware iSCSI adapters might need to be licensed. Otherwise, they will not appear in the vSphere Client or vSphere CLI. Contact your vendor for licensing information.

Establishing iSCSI Connections

In the ESXi context, the term target identifies a single storage unit that your host can access. The terms storage device and LUN describe a logical volume that represents storage space on a target. Typically, the terms device and LUN, in the ESXi context, mean a SCSI volume presented to your host from a storage target and available for formatting.

Different iSCSI storage vendors present storage to servers in different ways. Some vendors present multiple LUNs on a single target, while others present multiple targets with one LUN each. While the way the storage is used by ESXi is similar, the way the information is presented through administrative tools is different.

Figure 8-1. Target Compared to LUN Representations



Three LUNs are available in each of these configurations. In the first case, the host detects one target but that target has three LUNs that can be used. Each of the LUNs represents individual storage volume. In the second case, the host detects three different targets, each having one LUN.

Host-based iSCSI initiators establish connections to each target. Storage systems with a single target containing multiple LUNs have traffic to all the LUNs on a single connection. With a system that has three targets with one LUN each, a host uses separate connections to the three LUNs. This information is useful when you are trying to aggregate storage traffic on multiple connections from the host with multiple iSCSI HBAs, where traffic for one target can be set to a particular HBA, while traffic for another target can use a different HBA.

iSCSI Storage System Types

ESXi supports different storage systems and arrays.

The types of storage that your host supports include active-active, active-passive, and ALUA-compliant.

Active-active storage system — Allows access to the LUNs simultaneously through all the storage ports that are available without significant performance degradation. All the paths are active at all times, unless a path fails.

Active-passive storage system — A system in which one storage processor is actively providing access to a given LUN. The other processors act as backup for the LUN and can be actively providing access to other LUN I/O. I/O can be successfully sent only to an active port for a given LUN. If access through the active storage port fails, one of the passive storage processors can be activated by the servers accessing it.

Asymmetrical storage system — Supports Asymmetric Logical Unit Access (ALUA). ALUA-complaint storage systems provide different levels of access per port. ALUA allows hosts to determine the states of target ports and prioritize paths. The host uses some of the active paths as primary while others as secondary.

Virtual port storage system — Allows access to all available LUNs through a single virtual port. These are active-active storage devices, but hide their multiple connections though a single port. ESXi multipathing does not make multiple connections from a specific port to the storage by default. Some storage vendors supply session managers to establish and manage multiple connections to their storage. These storage systems handle port failover and connection balancing transparently. This is often referred to as transparent failover.

Discovery, Authentication, and Access Control

You can use several mechanisms to discover your storage and to limit access to it.

You must configure your host and the iSCSI storage system to support your storage access control policy.

Discovery

A discovery session is part of the iSCSI protocol, and it returns the set of targets you can access on an iSCSI storage system. The two types of discovery available on ESXi are dynamic and static. Dynamic discovery obtains a list of accessible targets from the iSCSI storage system, while static discovery can only try to access one particular target by target name and address.

For more information, see "Configuring Discovery Addresses for iSCSI Adapters," on page 81.

Authentication

iSCSI storage systems authenticate an initiator by a name and key pair. ESXi supports the CHAP protocol, which VMware recommends for your SAN implementation. To use CHAP authentication, the ESXi host and the iSCSI storage system must have CHAP enabled and have common credentials.

For information on enabling CHAP, see "Configuring CHAP Parameters for iSCSI Adapters," on page 82.

Access Control

Access control is a policy set up on the iSCSI storage system. Most implementations support one or more of three types of access control:

- By initiator name
- By IP address
- By the CHAP protocol

Only initiators that meet all rules can access the iSCSI volume.

Using only CHAP for access control can slow down rescans because the ESXi host can discover all targets, but then fails at the authentication step. iSCSI rescans work faster if the host discovers only the targets it can authenticate.

Error Correction

To protect the integrity of iSCSI headers and data, the iSCSI protocol defines error correction methods known as header digests and data digests.

Both parameters are disabled by default, but you can enable them. These digests pertain to, respectively, the header and SCSI data being transferred between iSCSI initiators and targets, in both directions.

Header and data digests check the end-to-end, noncryptographic data integrity beyond the integrity checks that other networking layers provide, such as TCP and Ethernet. They check the entire communication path, including all elements that can change the network-level traffic, such as routers, switches, and proxies.

The existence and type of the digests are negotiated when an iSCSI connection is established. When the initiator and target agree on a digest configuration, this digest must be used for all traffic between them.

Enabling header and data digests does require additional processing for both the initiator and the target and can affect throughput and CPU use performance.

NOTE Systems that use Intel Nehalem processors offload the iSCSI digest calculations, thus reducing the impact on performance.

For information on enabling header and data digests, see "Configuring Advanced Parameters for iSCSI," on page 86.

How Virtual Machines Access Data on an iSCSI SAN

ESXi stores a virtual machine's disk files within a VMFS datastore that resides on a SAN storage device. When virtual machine guest operating systems issue SCSI commands to their virtual disks, the SCSI virtualization layer translates these commands to VMFS file operations.

When a virtual machine interacts with its virtual disk stored on a SAN, the following process takes place:

1 When the guest operating system in a virtual machine reads or writes to SCSI disk, it issues SCSI commands to the virtual disk.

2 Device drivers in the virtual machine's operating system communicate with the virtual SCSI controllers.

3 The virtual SCSI controller forwards the command to the VMkernel.

4 The VMkernel performs the following tasks.

   a Locates the file, which corresponds to the guest virtual machine disk, in the VMFS volume.

   b Maps the requests for the blocks on the virtual disk to blocks on the appropriate physical device.

   c Sends the modified I/O request from the device driver in the VMkernel to the iSCSI initiator (hardware or software).

5 If the iSCSI initiator is a hardware iSCSI adapter (both independent or dependent), the adapter performs the following tasks.

   a Encapsulates I/O requests into iSCSI Protocol Data Units (PDUs).

   b Encapsulates iSCSI PDUs into TCP/IP packets.

   c Sends IP packets over Ethernet to the iSCSI storage system.

6 If the iSCSI initiator is a software iSCSI adapter, the following takes place.

   a The iSCSI initiator encapsulates I/O requests into iSCSI PDUs.

   b The initiator sends iSCSI PDUs through TCP/IP connections.

   c The VMkernel TCP/IP stack relays TCP/IP packets to a physical NIC.

   d The physical NIC sends IP packets over Ethernet to the iSCSI storage system.

7 Depending on which port the iSCSI initiator uses to connect to the network, Ethernet switches and routers carry the request to the storage device that the host wants to access.

Configuring iSCSI Adapters and Storage

9

Before ESXi can work with a SAN, you must set up your iSCSI adapters and storage.

To do this, you must first observe certain basic requirements and then follow best practices for installing and setting up hardware or software iSCSI adapters to access the SAN.

The following table lists the iSCSI adapters (vmhbas) that ESXi supports and indicates whether VMkernel networking configuration is required.

Table 9-1. Supported iSCSI adapters

iSCSI Adapter (vmhba)	Description	VMkernel Networking
Software	Uses standard NICs to connect your host to a remote iSCSI target on the IP network .	Required
Independent Hardware	Third-party adapter that offloads the iSCSI and network processing and management from your host.	Not required
Dependent Hardware	Third-party adapter that depends on VMware networking and iSCSI configuration and management interfaces.	Required

After you set up the iSCSI adapters, you can create a datastore on iSCSI storage. For details on how to create and manage datastores, see "Create a VMFS Datastore," on page 118.

This chapter includes the following topics:

- "ESXi iSCSI SAN Requirements," on page 68
- "ESXi iSCSI SAN Restrictions," on page 68
- "Setting LUN Allocations for iSCSI," on page 68
- "Network Configuration and Authentication," on page 69
- "Setting Up Independent Hardware iSCSI Adapters," on page 69
- "Configuring Dependent Hardware iSCSI Adapters," on page 70
- "Configuring Software iSCSI Adapter," on page 72
- "Setting Up iSCSI Network," on page 74
- "Using Jumbo Frames with iSCSI," on page 80
- "Configuring Discovery Addresses for iSCSI Adapters," on page 81
- "Configuring CHAP Parameters for iSCSI Adapters," on page 82

- "Configuring Advanced Parameters for iSCSI," on page 86
- "iSCSI Session Management," on page 87

ESXi iSCSI SAN Requirements

You must meet several requirements for your ESXi host to work properly with a SAN.

- Verify that your SAN storage hardware and firmware combinations are supported in conjunction with ESXi systems. For an up-to-date list, see *vSphere Compatibility Guide*.
- Configure your system to have only one VMFS datastore for each LUN.
- Unless you are using diskless servers, set up a diagnostic partition on a local storage. If you have diskless servers that boot from iSCSI SAN, see "General Boot from iSCSI SAN Recommendations," on page 97 for information about diagnostic partitions with iSCSI.
- Use RDMs for access to any raw disk. For information, see Chapter 14, "Raw Device Mapping," on page 135.
- Set the SCSI controller driver in the guest operating system to a large enough queue. For information on changing queue depth for iSCSI adapters and virtual machines, see *vSphere Troubleshooting*.
- On virtual machines running Microsoft Windows, increase the value of the SCSI `TimeoutValue` parameter to allow Windows to better tolerate delayed I/O resulting from path failover. For information, see "Set Timeout on Windows Guest OS," on page 157.

ESXi iSCSI SAN Restrictions

A number of restrictions exist when you use ESXi with an iSCSI SAN.

- ESXi does not support iSCSI-connected tape devices.
- You cannot use virtual-machine multipathing software to perform I/O load balancing to a single physical LUN.
- ESXi does not support multipathing when you combine independent hardware adapters with either software or dependent hardware adapters.

Setting LUN Allocations for iSCSI

When preparing your ESXi system to use iSCSI SAN storage you need to set LUN allocations.

Note the following points:

- Storage Provisioning. To ensure that the host recognizes LUNs at startup time, configure all iSCSI storage targets so that your host can access them and use them. Also, configure your host so that it can discover all available iSCSI targets.
- vMotion and VMware DRS. When you use vCenter Server and vMotion or DRS, make sure that the LUNs for the virtual machines are provisioned to all hosts. This configuration provides the greatest freedom in moving virtual machines.
- Active-active versus active-passive arrays. When you use vMotion or DRS with an active-passive SAN storage device, make sure that all hosts have consistent paths to all storage processors. Not doing so can cause path thrashing when a vMotion migration occurs.

  For active-passive storage arrays not listed in Storage/SAN Compatibility, VMware does not support storage-port failover. You must connect the server to the active port on the storage system. This configuration ensures that the LUNs are presented to the host.

Network Configuration and Authentication

Before your ESXi host can discover iSCSI storage, the iSCSI initiators must be configured and authentication might have to be set up.

- For software iSCSI and dependent hardware iSCSI, networking for the VMkernel must be configured. You can verify the network configuration by using the `vmkping` utility. For independent hardware iSCSI, network parameters, such as IP address, subnet mask, and default gateway must be configured on the HBA.
- Check and change the default initiator name if necessary.
- The dynamic discovery address or static discovery address and target name of the storage system must be set. For software iSCSI and dependent hardware iSCSI, the address should be pingable using `vmkping`.
- For CHAP authentication, enable it on the initiator and the storage system side. After authentication is enabled, it applies for all of the targets that are not yet discovered, but does not apply to targets that are already discovered. After the discovery address is set, the new targets discovered are exposed and can be used at that point.

For details on how to use the `vmkping` command, search the VMware Knowledge Base.

Setting Up Independent Hardware iSCSI Adapters

An independent hardware iSCSI adapter is a specialized third-party adapter capable of accessing iSCSI storage over TCP/IP. This iSCSI adapter handles all iSCSI and network processing and management for your ESXi system.

The setup and configuration process for the independent hardware iSCSI adapters involves these steps:

1 Check whether the adapter needs to be licensed.

  See your vendor documentation.

2 Install the adapter.

  For installation information and information on firmware updates, see vendor documentation.

3 Verity that the adapter is installed correctly.

  See "View Independent Hardware iSCSI Adapters," on page 69.

4 Configure discovery information.

  See "Configuring Discovery Addresses for iSCSI Adapters," on page 81.

5 (Optional) Configure CHAP parameters.

  See "Configuring CHAP Parameters for iSCSI Adapters," on page 82.

View Independent Hardware iSCSI Adapters

View an independent hardware iSCSI adapter to verify that it is correctly installed and ready for configuration.

After you install an independent hardware iSCSI adapter, it appears on the list of storage adapters available for configuration. You can view its properties.

Prerequisites

Required privilege: **Host.Configuration.Storage Partition Configuration**

Procedure

1 Log in to the vSphere Client, and select a host from the inventory panel.

2 Click the **Configuration** tab and click **Storage Adapters** in the Hardware panel.

If it is installed, the hardware iSCSI adapter appears on the list of storage adapters.

3 Select the adapter to view.

The default details for the adapter appear, including the model, iSCSI name, iSCSI alias, IP address, and target and paths information.

4 Click **Properties**.

The iSCSI Initiator Properties dialog box appears. The **General** tab displays additional characteristics of the adapter.

You can now configure your independent hardware adapter or change its default characteristics.

Change Name and IP Address for Independent Hardware iSCSI Adapters

When you configure your independent hardware iSCSI adapters, make sure that their names and IP addresses are formatted properly.

Prerequisites

Required privilege: **Host .Configuration.Storage Partition Configuration**

Procedure

1 Access the iSCSI Initiator Properties dialog box.

2 Click **Configure**.

3 To change the default iSCSI name for your adapter, enter the new name.

Make sure the name you enter is worldwide unique and properly formatted or some storage devices might not recognize the iSCSI adapter.

4 (Optional) Enter the iSCSI alias.

The alias is a name that you use to identify the independent hardware iSCSI adapter.

5 Change the default IP settings.

You must change the default IP settings so that they are configured properly for the IP SAN. Work with your network administrator to determine the IP setting for the HBA.

6 Click **OK** to save your changes.

If you change the iSCSI name, it will be used for new iSCSI sessions. For existing sessions, new settings will not be used until logout and re-login.

Configuring Dependent Hardware iSCSI Adapters

A dependent hardware iSCSI adapter is a third-party adapter that depends on VMware networking, and iSCSI configuration and management interfaces provided by VMware.

An example of a dependent iSCSI adapter is a Broadcom 5709 NIC. When installed on a host, it presents its two components, a standard network adapter and an iSCSI engine, to the same port. The iSCSI engine appears on the list of storage adapters as an iSCSI adapter (vmhba). Although the iSCSI adapter is enabled by default, to make it functional, you must first connect it, through a virtual VMkernel interface, to a physical network adapter (vmnic) associated with it. You can then configure the iSCSI adapter.

After you configure the dependent hardware iSCSI adapter, the discovery and authentication data are passed through the network connection, while the iSCSI traffic goes through the iSCSI engine, bypassing the network.

The entire setup and configuration process for the dependent hardware iSCSI adapters involves these steps:

1 View the dependent hardware adapters.

See "View Dependent Hardware iSCSI Adapters," on page 71.

If your dependent hardware adapters do not appear on the list of storage adapters, check whether they need to be licensed. See your vendor documentation.

2 Determine the association between the dependent hardware adapters and physical NICs.

See "Determine Association Between iSCSI and Network Adapters," on page 72

Make sure to note the names of the corresponding physical NICs. For example, the vmhba33 adapter corresponds to vmnic1 and vmhba34 corresponds to vmnic2.

3 Configure networking for iSCSI.

See "Setting Up iSCSI Network," on page 74.

Configuring the network involves creating a VMkernel interface for each physical network adapter and associating the interface with an appropriate iSCSI adapter.

4 Configure discovery information.

See "Configuring Discovery Addresses for iSCSI Adapters," on page 81.

5 (Optional) Configure CHAP parameters.

See "Configuring CHAP Parameters for iSCSI Adapters," on page 82.

Dependent Hardware iSCSI Considerations

When you use dependent hardware iSCSI adapters with ESXi, certain considerations apply.

- When you use any dependent hardware iSCSI adapter, performance reporting for a NIC associated with the adapter might show little or no activity, even when iSCSI traffic is heavy. This behavior occurs because the iSCSI traffic bypasses the regular networking stack.
- If you use a third-party virtual switch, for example Cisco Nexus 1000V DVS, disable automatic pinning. Use manual pinning instead, making sure to connect a VMkernel adapter (vmk) to an appropriate physical NIC (vmnic). For information, refer to your virtual switch vendor documentation.
- The Broadcom iSCSI adapter performs data reassembly in hardware, which has a limited buffer space. When you use the Broadcom iSCSI adapter in a congested network or under heavy load, enable flow control to avoid performance degradation.

  Flow control manages the rate of data transmission between two nodes to prevent a fast sender from overrunning a slow receiver. For best results, enable flow control at the end points of the I/O path, at the hosts and iSCSI storage systems.
- Broadcom iSCSI adapters do not support IPv6 and Jumbo Frames.

View Dependent Hardware iSCSI Adapters

View a dependent hardware iSCSI adapter to verify that it is correctly loaded.

If the dependent hardware adapter does not appear on the list of storage adapters, check whether it needs to be licensed. See your vendor documentation.

Prerequisites

Required privilege: **Host.Configuration.Storage Partition Configuration**

Procedure

1 Log in to the vSphere Client, and select a host from the inventory panel.

2 Click the **Configuration** tab and click **Storage Adapters** in the Hardware panel.

If it is installed, the dependent hardware iSCSI adapter appears on the list of storage adapters under such category as, for example, Broadcom iSCSI Adapter.

3 Select the adapter to view and click **Properties**.

The iSCSI Initiator Properties dialog box opens. It displays the default details for the adapter, including the iSCSI name, iSCSI alias, and the status.

4 (Optional) To change the default iSCSI name, click **Configure**.

What to do next

Although the dependent iSCSI adapter is enabled by default, to make it functional, you must set up networking for the iSCSI traffic and bind the adapter to the appropriate VMkernel iSCSI port. You then configure discovery addresses and CHAP parameters.

Determine Association Between iSCSI and Network Adapters

You create network connections to bind dependent iSCSI and network adapters. To create the connections correctly, you must determine the name of the physical NIC with which the dependent hardware iSCSI adapter is associated.

Prerequisites

Required privilege: **Host.Configuration.Storage Partition Configuration**

Procedure

1 In the iSCSI Initiator Properties dialog box, click the **Network Configuration** tab.

2 Click **Add**.

The network adapter, for example vmnic2, that corresponds to the dependent iSCSI adapter is listed.

What to do next

You must bind the associated dependent hardware iSCSI and network adapters by creating the network connections.

Configuring Software iSCSI Adapter

With the software-based iSCSI implementation, you can use standard NICs to connect your host to a remote iSCSI target on the IP network. The software iSCSI adapter that is built into ESXi facilitates this connection by communicating with the physical NICs through the network stack.

Before you can use the software iSCSI adapter, you must set up networking, activate the adapter, and configure parameters such as discovery addresses and CHAP.

NOTE Designate a separate network adapter for iSCSI. Do not use iSCSI on 100Mbps or slower adapters.

The software iSCSI adapter configuration workflow includes these steps:

1 Activate the software iSCSI adapter.

See "Activate the Software iSCSI Adapter," on page 73.

2 Configure networking for iSCSI.

See "Setting Up iSCSI Network," on page 74.

Configuring the network involves creating a VMkernel interface for each physical network adapter that you use for iSCSI and associating all interfaces with the software iSCSI adapter.

3 If needed, enable Jumbo Frames.

See "Enable Jumbo Frames for iSCSI," on page 80.

4 Configure discovery information.

See "Configuring Discovery Addresses for iSCSI Adapters," on page 81.

5 (Optional) Configure CHAP parameters.

See "Configuring CHAP Parameters for iSCSI Adapters," on page 82.

Activate the Software iSCSI Adapter

You must activate your software iSCSI adapter so that your host can use it to access iSCSI storage.

You can activate only one software iSCSI adapter.

Prerequisites

Required privilege: **Host.Configuration.Storage Partition Configuration**

NOTE If you boot from iSCSI using the software iSCSI adapter, the adapter is enabled and the network configuration is created at the first boot. If you disable the adapter, it is reenabled each time you boot the host.

Procedure

1 Log in to the vSphere Client, and select a host from the inventory panel.

2 Click the **Configuration** tab and click **Storage Adapters** in the Hardware panel.

3 Click **Add** and select **Software iSCSI Adapter**.

The software iSCSI adapters appears on the list of storage adapters.

4 Select the iSCSI adapter from the list and click **Properties**.

5 Click **Configure**.

6 Make sure that the adapter is enabled and click **OK**.

After enabling the adapter, the host assigns the default iSCSI name to it. If you change the default name, follow iSCSI naming conventions.

After you activate the adapter, you can disable it, but you cannot remove it from the list of storage adapters.

Disable the Software iSCSI Adapter

Use the vSphere Client to disable the software iSCSI adapter if you do not need it.

NOTE If you disable the adapter that is used for software iSCSI boot, the adapter is reenabled each time you boot the host.

Prerequisites

Required privilege: **Host.Configuration.Storage Partition Configuration**

Procedure

1 Log in to the vSphere Client, and select a host from the inventory panel.

2 Click the **Configuration** tab and click **Storage Adapters** in the Hardware panel.

3 Select the software iSCSI adapter from the list of storage adapters and click **Properties**.

4 Click **Configure**.

5 To disable the adapter, deselect **Enabled** and click **OK**.

6 Reboot the host.

After reboot, the adapter no longer appears on the list of storage adapters.

The status indicates that the adapter is disabled.

Setting Up iSCSI Network

Software and dependent hardware iSCSI adapters depend on VMkernel networking. If you use the software or dependent hardware iSCSI adapters, you must configure connections for the traffic between the iSCSI component and the physical network adapters.

Configuring the network connection involves creating a virtual VMkernel interface for each physical network adapter and associating the interface with an appropriate iSCSI adapter.

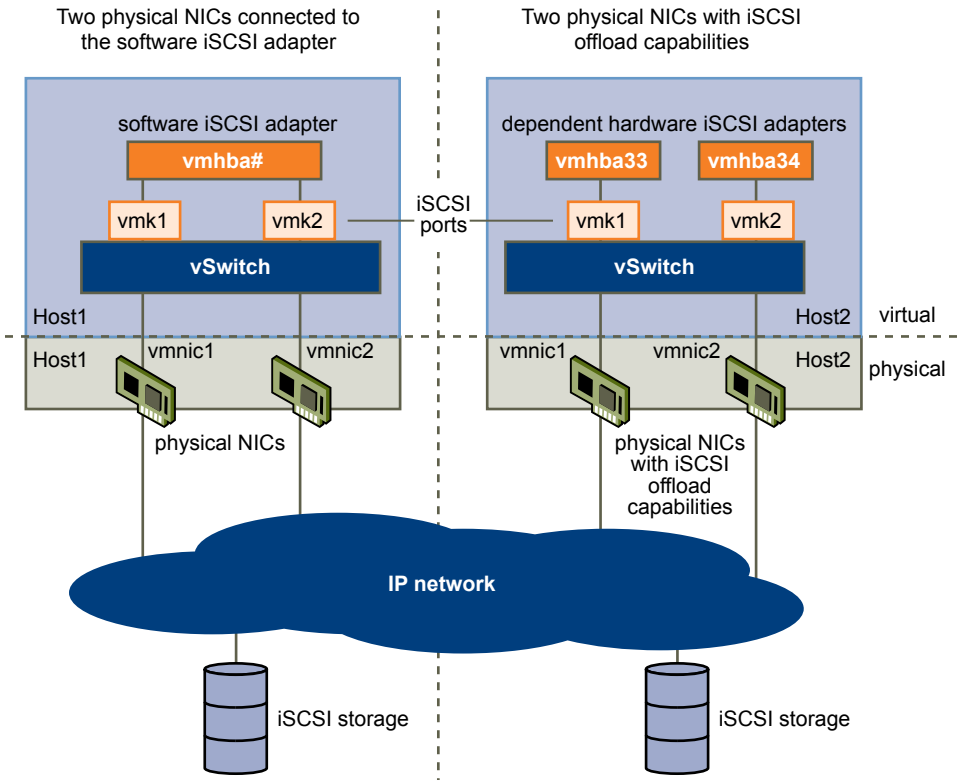
Multiple Network Adapters in iSCSI Configuration

If your host has more than one physical network adapter for software and dependent hardware iSCSI, use the adapters for multipathing.

You can connect the software iSCSI adapter with any physical NICs available on your host. The dependent iSCSI adapters must be connected only with their own physical NICs.

NOTE Physical NICs must be on the same subnet as the iSCSI storage system they connect to.

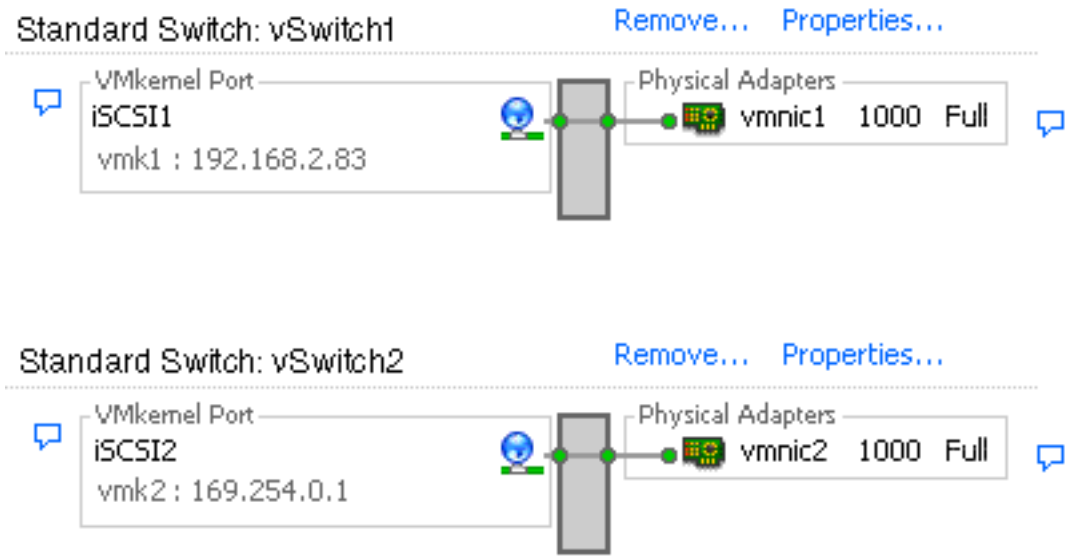
Figure 9-1. Networking with iSCSI



The iSCSI adapter and physical NIC connect through a virtual VMkernel adapter, also called virtual network adapter or VMkernel port. You create a VMkernel adapter (vmk) on a vSphere switch (vSwitch) using 1:1 mapping between each virtual and physical network adapter.

One way to achieve the 1:1 mapping when you have multiple NICs, is to designate a separate vSphere switch for each virtual-to-physical adapter pair. The following examples show configurations that use vSphere standard switches, but you can use distributed switches as well. For more information about vSphere distributed switches, see the *vSphere Networking* documentation.

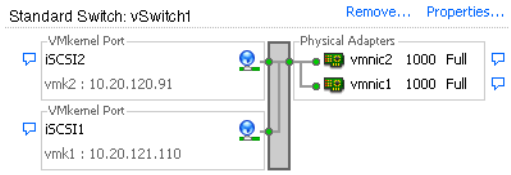
Figure 9-2. 1:1 adapter mapping on separate vSphere standard switches



NOTE If you use separate vSphere switches, you must connect them to different IP subnets. Otherwise, VMkernel adapters might experience connectivity problems and the host will fail to discover iSCSI LUNs.

An alternative is to add all NICs and VMkernel adapters to a single vSphere standard switch. In this case, you must override the default network setup and make sure that each VMkernel adapter maps to only one corresponding active physical adapter.

Figure 9-3. 1:1 adapter mapping on a single vSphere standard switch



The following table summarises the iSCSI networking configuration discussed in this topic.

Table 9-2. Networking configuration for iSCSI

iSCSI Adapters	VMkernel Adapters (Ports)	Physical Adapters (NICs)
Software iSCSI		
vmhba32	vmk1	vmnic1
	vmk2	vmnic2
Dependent Hardware iSCSI		
vmhba33	vmk1	vmnic1
vmhba34	vmk2	vmnic2

Create Network Connections for iSCSI

Configure connections for the traffic between the software or dependent hardware iSCSI adapters and the physical network adapters.

The following tasks discuss the iSCSI network configuration with a vSphere standard switch.

If you use a vSphere distributed switch with multiple uplink ports, for port binding, create a separate distributed port group per each physical NIC. Then set the team policy so that each distributed port group has only one active uplink port. For detailed information on vSphere distributed switches, see the *vSphere Networking* documentation.

Procedure

1 Create a Single VMkernel Adapter for iSCSI on page 76

You must connect the VMkernel, which runs services for iSCSI storage, to a physical network adapter.

2 Create Additional VMkernel Adapters for iSCSI on page 77

Use this task if you have two or more physical network adapters for iSCSI and you want to connect all of your NICs to a single vSphere standard switch. In this task, you add NICs and VMkernel adapters to an existing vSphere standard switch.

3 Change Port Group Policy for iSCSI VMkernel Adapters on page 78

If you use a single vSphere standard switch to connect VMkernel to multiple network adapters, change the port group policy, so that it is compatible with the iSCSI network requirements.

4 Bind iSCSI Adapters with VMkernel Adapters on page 78

Bind an iSCSI adapter with a VMkernel adapter.

Create a Single VMkernel Adapter for iSCSI

You must connect the VMkernel, which runs services for iSCSI storage, to a physical network adapter.

Procedure

1 Log in to the vSphere Client, and select a host from the inventory panel.

2 Click the **Configuration** tab and click **Networking**.

3 In the vSphere Standard Switch view, click **Add Networking**.

4 Select **VMkernel** and click **Next**.

5 Select **Create a vSphere standard switch** to create a new standard switch.

6 Select a NIC to use for iSCSI traffic.

IMPORTANT If you are creating a VMkernel interface for the dependent hardware iSCSI adapter, select the NIC that corresponds to the iSCSI component. See "Determine Association Between iSCSI and Network Adapters," on page 72.

7 Click **Next**.

8 Enter a network label.

A network label is a friendly name that identifies the VMkernel adapter that you are creating, for example, iSCSI.

9 Click **Next**.

10 Specify the IP settings and click **Next**.

11 Review the information and click **Finish**.

You created the virtual VMkernel adapter for a physical network adapter on your host.

What to do next

If your host has one physical network adapter for iSCSI traffic, you must bind the virtual adapter that you created to the iSCSI adapter.

If you have multiple network adapters, create additional VMkernel adapters and then perform iSCSI binding. The number of virtual adapters must correspond to the number of physical adapters on the host.

Create Additional VMkernel Adapters for iSCSI

Use this task if you have two or more physical network adapters for iSCSI and you want to connect all of your NICs to a single vSphere standard switch. In this task, you add NICs and VMkernel adapters to an existing vSphere standard switch.

Prerequisites

You must create a vSphere standard switch that maps an iSCSI VMkernel adapter to a single physical NIC designated for iSCSI traffic.

Procedure

1 Log in to the vSphere Client, and select a host from the inventory panel.

2 Click the **Configuration** tab and click **Networking**.

3 Select the vSphere standard switch that you use for iSCSI and click **Properties**.

4 Connect additional network adapters to the standard switch.

   a In the standard switch Properties dialog box, click the **Network Adapters** tab and click **Add**.

   b Select one or more NICs from the list and click **Next**.

   With dependent hardware iSCSI adapters, select only those NICs that have a corresponding iSCSI component.

   c Review the information on the Adapter Summary page and click **Finish**.

   The list of network adapters reappears, showing the network adapters that the vSphere standard switch now claims.

5 Create iSCSI VMkernel adapters for all NICs that you added.

   The number of VMkernel interfaces must correspond to the number of NICs on the vSphere standard switch.

   a In the standard switch Properties dialog box, click the **Ports** tab and click **Add**.

   b Select **VMkernel** and click **Next**.

   c Under **Port Group Properties**, enter a network label, for example iSCSI, and click **Next**.

   d Specify the IP settings and click **Next**.

   When you enter the subnet mask, make sure that the NIC is set to the subnet of the storage system it connects to.

   e Review the information and click **Finish**.

CAUTION If the NIC you use with your iSCSI adapter, either software or dependent hardware, is not in the same subnet as your iSCSI target, your host cannot establish sessions from this network adapter to the target.

What to do next

Change the network policy for all VMkernel adapters, so that it is compatible with the network binding requirements. You can then bind the iSCSI VMkernel adapters to the software iSCSI or dependent hardware iSCSI adapters.

Change Port Group Policy for iSCSI VMkernel Adapters

If you use a single vSphere standard switch to connect VMkernel to multiple network adapters, change the port group policy, so that it is compatible with the iSCSI network requirements.

By default, for each virtual adapter on the vSphere standard switch, all network adapters appear as active. You must override this setup, so that each VMkernel interface maps to only one corresponding active NIC. For example, vmk1 maps to vmnic1, vmk2 maps to vmnic2, and so on.

Prerequisites

Create a vSphere standard switch that connects VMkernel with physical network adapters designated for iSCSI traffic. The number of VMkernel adapters must correspond to the number of physical adapters on the vSphere standard switch.

Procedure

1 Log in to the vSphere Client and select the host from the inventory panel.

2 Click the **Configuration** tab and click **Networking**.

3 Select the vSphere standard switch that you use for iSCSI and click **Properties**.

4 On the **Ports** tab, select an iSCSI VMkernel adapter and click **Edit**.

5 Click the **NIC Teaming** tab and select **Override switch failover order**.

6 Designate only one physical adapter as active and move all remaining adapters to the **Unused Adapters** category.

7 Repeat Step 4 through Step 6 for each iSCSI VMkernel interface on the vSphere standard switch.

What to do next

After you perform this task, bind the virtual VMkernel adapters to the software iSCSI or dependent hardware iSCSI adapters.

Bind iSCSI Adapters with VMkernel Adapters

Bind an iSCSI adapter with a VMkernel adapter.

Prerequisites

Create a virtual VMkernel adapter for each physical network adapter on your host. If you use multiple VMkernel adapters, set up the correct network policy.

Required privilege: **Host.Configuration.Storage Partition Configuration**

Procedure

1 Log in to the vSphere Client, and select a host from the inventory panel.

2 Click the **Configuration** tab, and click **Storage Adapters** in the Hardware panel.

The list of available storage adapters appears.

3 Select the software or dependent iSCSI adapter to configure and click **Properties**.

4 In the iSCSI Initiator Properties dialog box, click the **Network Configuration** tab.

5 Click **Add** and select a VMkernel adapter to bind with the iSCSI adapter.

You can bind the software iSCSI adapter to one or more VMkernel adapters. For a dependent hardware iSCSI adapter, only one VMkernel interface associated with the correct physical NIC is available.

6 Click **OK**.

The network connection appears on the list of VMkernel port bindings for the iSCSI adapter.

7 Verify that the network policy for the connection is compliant with the binding requirements.

Managing iSCSI Network

Special consideration apply to network adapters, both physical and VMkernel, that are associated with an iSCSI adapter.

After you create network connections for iSCSI, an iSCSI indicator on a number of Networking dialog boxes becomes enabled. This indicator shows that a particular virtual or physical network adapter is iSCSI-bound. To avoid disruptions in iSCSI traffic, follow these guidelines and considerations when managing iSCSI-bound virtual and physical network adapters:

- Make sure that the VMkernel network adapters are assigned addresses on the same subnet as the iSCSI storage portal they connect to.
- iSCSI adapters using VMkernel adapters are not able to connect to iSCSI ports on different subnets, even if those ports are discovered by the iSCSI adapters.
- When using separate vSphere switches to connect physical network adapters and VMkernel adapters, make sure that the vSphere switches connect to different IP subnets.
- If you migrate VMkernel adapters to a different vSphere switch, move associated physical adapters.
- Do not make configuration changes to iSCSI-bound VMkernel adapters or physical network adapters.
- Do not make changes that might break association of VMkernel adapters and physical network adapters. You can break the association if you remove one of the adapters or the vSphere switch that connects them, or change the 1:1 network policy for their connection.

iSCSI Network Troubleshooting

A warning sign indicates non-compliant port group policy for an iSCSI-bound VMkernel adapter.

Problem

The VMkernel adapter's port group policy is considered non-compliant in the following cases:

- The VMkernel adapter is not connected to an active physical network adapter.
- The VMkernel adapter is connected to more than one physical network adapter.
- The VMkernel adapter is connected to one or more standby physical adapters.
- The active physical adapter is changed.

Solution

Follow the steps in "Change Port Group Policy for iSCSI VMkernel Adapters," on page 78 to set up the correct network policy for the iSCSI-bound VMkernel adapter.

Using Jumbo Frames with iSCSI

ESXi supports the use of Jumbo Frames with iSCSI.

Jumbo Frames are Ethernet frames with the size that exceeds 1500 Bytes. The maximum transmission unit (MTU) parameter is typically used to measure the size of Jumbo Frames. ESXi allows Jumbo Frames with the MTU up to 9000 Bytes.

When you use Jumbo Frames for iSCSI traffic, the following considerations apply:

- The network must support Jumbo Frames end-to-end for Jumbo Frames to be effective.
- Check with your vendors to ensure your physical NICs support Jumbo Frames.
- To set up and verify physical network switches for Jumbo Frames, consult your vendor documentation.

The following table explains the level of support that ESXi provides to Jumbo Frames.

Table 9-3. Support of Jumbo Frames

Type of iSCSI Adapters	Jumbo Frames Support
Software iSCSI	Supported
Dependent Hardware iSCSI	Check with Vendor
Independent Hardware iSCSI	Not Supported

Enable Jumbo Frames for iSCSI

Use the vSphere Client to enable Jumbo Frames for each vSphere standard switch and VMkernel adapter designated for iSCSI traffic.

Enable Jumbo Frames on the Standard switch and VMkernel adapter by changing the maximum transmission units (MTU) parameter.

Procedure

1 Log in to the vSphere Client, and select a host from the inventory panel.

2 Click the **Configuration** tab and click **Networking**.

3 Click **Properties** for the standard switch you use for iSCSI traffic.

4 On the Ports tab, select the standard switch and click **Edit**.

5 Set the MTU parameter for the standard switch, and click **OK**.

This step sets the MTU for all physical NICs on that standard switch. The MTU value should be set to the largest MTU size among all NICs connected to the standard switch.

6 On the Ports tab, select the VMkernel adapter and click **Edit**.

7 Set the MTU to match the value configured on the standard switch, and click **OK**.

Configuring Discovery Addresses for iSCSI Adapters

Set up target discovery addresses so that the iSCSI adapter can determine which storage resource on the network is available for access.

The ESXi system supports these discovery methods:

Dynamic Discovery — Also known as SendTargets discovery. Each time the initiator contacts a specified iSCSI server, the initiator sends the SendTargets request to the server. The server responds by supplying a list of available targets to the initiator. The names and IP addresses of these targets appear on the **Static Discovery** tab. If you remove a static target added by dynamic discovery, the target might be returned to the list the next time a rescan happens, the HBA is reset, or the host is rebooted.

Static Discovery — The initiator does not have to perform any discovery. The initiator has a list of targets it can contact and uses their IP addresses and target names to communicate with them.

Set Up Dynamic Discovery

With Dynamic Discovery, each time the initiator contacts a specified iSCSI storage system, it sends the SendTargets request to the system. The iSCSI system responds by supplying a list of available targets to the initiator.

Required privilege: **Host.Configuration.Storage Partition Configuration**

When you set up Dynamic Discovery, you can only add a new iSCSI system. You cannot change the IP address, DNS name, or port number of an existing iSCSI system. To make changes, delete the existing system and add a new one.

Procedure

1 Log in to the vSphere Client, and select a host from the inventory panel.

2 Click the **Configuration** tab, and click **Storage Adapters** in the Hardware panel.

The list of available storage adapters appears.

3 Click the **Configuration** tab, and click **Storage Adapters** in the Hardware panel.

The list of available storage adapters appears.

4 Select the iSCSI initiator to configure, and click **Properties**.

5 Click the **Dynamic Discovery** tab.

6 To add an address for the SendTargets discovery, click **Add**.

7 Type the IP address or DNS name of the storage system and click **OK**.

After your host establishes the SendTargets session with this system, any newly discovered targets appear in the Static Discovery list.

8 To delete a specific SendTargets server, select it and click **Remove**.

After you remove a SendTargets server, it might still appear in the Inheritance field as the parent of static targets. This entry indicates where the static targets were discovered and does not affect the functionality.

What to do next

After configuring Dynamic Discovery for your iSCSI adapter, rescan the adapter.

Set Up Static Discovery

With iSCSI initiators, in addition to the dynamic discovery method, you can use static discovery and manually enter information for the targets.

Required privilege: **Host.Configuration.Storage Partition Configuration**

When you set up Static Discovery, you can only add iSCSI targets. You cannot change the IP address, DNS name, iSCSI target name, or port number of an existing target. To make changes, remove the existing target and add a new one.

Procedure

1 Log in to the vSphere Client, and select a host from the inventory panel.

2 Click the **Configuration** tab, and click **Storage Adapters** in the Hardware panel.

The list of available storage adapters appears.

3 Select the iSCSI initiator to configure and click **Properties**.

4 Click the **Static Discovery** tab.

The tab displays all dynamically discovered targets and any static targets already entered.

5 To add a target, click **Add** and enter the target's information.

6 To delete a specific target, select the target and click **Remove**.

What to do next

After configuring Static Discovery for your iSCSI adapter, rescan the adapter.

Configuring CHAP Parameters for iSCSI Adapters

Because the IP networks that the iSCSI technology uses to connect to remote targets do not protect the data they transport, you must ensure security of the connection. One of the protocols that iSCSI implements is the Challenge Handshake Authentication Protocol (CHAP), which verifies the legitimacy of initiators that access targets on the network.

CHAP uses a three-way handshake algorithm to verify the identity of your host and, if applicable, of the iSCSI target when the host and target establish a connection. The verification is based on a predefined private value, or CHAP secret, that the initiator and target share.

ESXi supports CHAP authentication at the adapter level. In this case, all targets receive the same CHAP name and secret from the iSCSI initiator. For software and dependent hardware iSCSI adapters, ESXi also supports per-target CHAP authentication, which allows you to configure different credentials for each target to achieve greater level of security.

Choosing CHAP Authentication Method

ESXi supports one-way CHAP for all types of iSCSI initiators, and mutual CHAP for software and dependent hardware iSCSI.

Before configuring CHAP, check whether CHAP is enabled at the iSCSI storage system and check the CHAP authentication method the system supports. If CHAP is enabled, enable it for your initiators, making sure that the CHAP authentication credentials match the credentials on the iSCSI storage.

ESXi supports the following CHAP authentication methods:

One-way CHAP In one-way CHAP authentication, also called unidirectional, the target authenticates the initiator, but the initiator does not authenticate the target.

Mutual CHAP In mutual CHAP authentication, also called bidirectional, an additional level of security enables the initiator to authenticate the target. VMware supports this method for software and dependent hardware iSCSI adapters only.

For software and dependent hardware iSCSI adapters, you can set one-way CHAP and mutual CHAP for each initiator or at the target level. Independent hardware iSCSI supports CHAP only at the initiator level.

When you set the CHAP parameters, specify a security level for CHAP.

NOTE When you specify the CHAP security level, how the storage array responds depends on the array's CHAP implementation and is vendor specific. For example, when you select `Use CHAP unless prohibited by target`, some storage arrays use CHAP in response, while others do not. For information on CHAP authentication behavior in different initiator and target configurations, consult the array documentation.

Table 9-4. CHAP Security Level

CHAP Security Level	Description	Supported
Do not use CHAP	The host does not use CHAP authentication. Select this option to disable authentication if it is currently enabled.	Software iSCSI Dependent hardware iSCSI Independent hardware iSCSI
Do not use CHAP unless required by target	The host prefers a non-CHAP connection, but can use a CHAP connection if required by the target.	Software iSCSI Dependent hardware iSCSI
Use CHAP unless prohibited by target	The host prefers CHAP, but can use non-CHAP connections if the target does not support CHAP.	Software iSCSI Dependent hardware iSCSI Independent hardware iSCSI
Use CHAP	The host requires successful CHAP authentication. The connection fails if CHAP negotiation fails.	Software iSCSI Dependent hardware iSCSI

Set Up CHAP Credentials for iSCSI Initiator

You can set up all targets to receive the same CHAP name and secret from the iSCSI initiator at the initiator level. By default, all discovery addresses or static targets inherit CHAP parameters that you set up at the initiator level.

The CHAP name should not exceed 511 alphanumeric characters and the CHAP secret should not exceed 255 alphanumeric characters. Some adapters, for example the QLogic adapter, might have lower limits, 255 for the CHAP name and 100 for the CHAP secret.

Prerequisites

- Before setting up CHAP parameters for software or dependent hardware iSCSI, determine whether to configure one-way or mutual CHAP. Independent hardware iSCSI adapters do not support mutual CHAP.
  - In one-way CHAP, the target authenticates the initiator.
  - In mutual CHAP, both the target and the initiator authenticate each other. Use different secrets for CHAP and mutual CHAP.

  When you configure CHAP parameters, verify that they match the parameters on the storage side.

- Required privilege: **Host.Configuration.Storage Partition Configuration**

Procedure

1 Access the iSCSI Initiator Properties dialog box.

2 On the **General** tab, click **CHAP**.

3 To configure one-way CHAP, under CHAP specify the following:

   a Select the CHAP security level.

      - Do not use CHAP unless required by target (software and dependent hardware iSCSI only)
      - Use CHAP unless prohibited by target
      - Use CHAP (software and dependent hardware iSCSI only). To configure mutual CHAP, you must select this option.

   b Specify the CHAP name.

      Make sure that the name you specify matches the name configured on the storage side.

      - To set the CHAP name to the iSCSI initiator name, select **Use initiator name**.
      - To set the CHAP name to anything other than the iSCSI initiator name, deselect **Use initiator name** and type a name in the **Name** text box.

   c Enter a one-way CHAP secret to be used as part of authentication. Use the same secret that you enter on the storage side.

4 To configure mutual CHAP, first configure one-way CHAP by following the directions in Step 3.

   Make sure to select **Use CHAP** as an option for one-way CHAP. Then, specify the following under **Mutual CHAP**:

   a Select **Use CHAP**.

   b Specify the mutual CHAP name.

   c Enter the mutual CHAP secret. Make sure to use different secrets for the one-way CHAP and mutual CHAP.

5 Click **OK**.

6 Rescan the initiator.

If you change the CHAP or mutual CHAP parameters, they are used for new iSCSI sessions. For existing sessions, new settings are not used until you log out and log in again.

Set Up CHAP Credentials for Target

For software and dependent hardware iSCSI adapters, you can configure different CHAP credentials for each discovery address or static target.

When configuring CHAP parameters, make sure that they match the parameters on the storage side. The CHAP name should not exceed 511 and the CHAP secret 255 alphanumeric characters.

Required privilege: **Host.Configuration.Storage Partition Configuration**

Prerequisites

Before setting up CHAP parameters for software and dependent hardware iSCSI, determine whether to configure one-way or mutual CHAP.

- In one-way CHAP, the target authenticates the initiator.

- In mutual CHAP, both the target and initiator authenticate each other. Make sure to use different secrets for CHAP and mutual CHAP.

Procedure

1 Access the iSCSI Initiator Properties dialog box.

2 Select either **Dynamic Discovery** tab or **Static Discovery** tab.

3 From the list of available targets, select a target you want to configure and click **Settings > CHAP**.

4 Configure one-way CHAP in the CHAP area.

   a Deselect **Inherit from parent**.

   b Select one of the following options:

      - Do not use CHAP unless required by target
      - Use CHAP unless prohibited by target
      - Use CHAP. To be able to configure mutual CHAP, you must select this option.

   c Specify the CHAP name.

      Make sure that the name you specify matches the name configured on the storage side.

      - To set the CHAP name to the iSCSI initiator name, select **Use initiator name**.
      - To set the CHAP name to anything other than the iSCSI initiator name, deselect **Use initiator name** and enter a name in the **Name** field.

   d Enter a one-way CHAP secret to be used as part of authentication. Make sure to use the same secret that you enter on the storage side.

5 To configure mutual CHAP, first configure one-way CHAP by following directions in Step 4.

   Make sure to select **Use CHAP** as an option for one-way CHAP. Then, specify the following in the Mutual CHAP area:

   a Deselect **Inherit from parent**.

   b Select **Use CHAP**.

   c Specify the mutual CHAP name.

   d Enter the mutual CHAP secret. Make sure to use different secrets for the one-way CHAP and mutual CHAP.

6 Click **OK**.

7 Rescan the initiator.

If you change the CHAP or mutual CHAP parameters, they are used for new iSCSI sessions. For existing sessions, new settings are not used until you log out and login again.

Disable CHAP

You can disable CHAP if your storage system does not require it.

If you disable CHAP on a system that requires CHAP authentication, existing iSCSI sessions remain active until you reboot your host, end the session through the command line, or the storage system forces a logout. After the session ends, you can no longer connect to targets that require CHAP.

Required privilege: **Host.Configuration.Storage Partition Configuration**

Procedure

1 Open the CHAP Credentials dialog box.

2 For software and dependent hardware iSCSI adapters, to disable just the mutual CHAP and leave the one-way CHAP, select **Do not use CHAP** in the Mutual CHAP area.

3 To disable one-way CHAP, select **Do not use CHAP** in the CHAP area.

The mutual CHAP, if set up, automatically turns to **Do not use CHAP** when you disable the one-way CHAP.

4 Click **OK**.

Configuring Advanced Parameters for iSCSI

You might need to configure additional parameters for your iSCSI initiators. For example, some iSCSI storage systems require ARP (Address Resolution Protocol) redirection to move iSCSI traffic dynamically from one port to another. In this case, you must activate ARP redirection on your host.

The following table lists advanced iSCSI parameters that you can configure using the vSphere Client. In addition, you can use the vSphere CLI commands to configure some of the advanced parameters. For information, see the *Getting Started with vSphere Command-Line Interfaces* documentation.

Do not make any changes to the advanced iSCSI settings unless you are working with the VMware support team or have thorough information about the values to provide for the settings.

Table 9-5. Additional Parameters for iSCSI Initiators

Advanced Parameter	Description	Configurable On
Header Digest	Increases data integrity. When header digest is enabled, the system performs a checksum over each iSCSI Protocol Data Unit's (PDU's) header part and verifies using the CRC32C algorithm.	Software iSCSI Dependent Hardware iSCSI
Data Digest	Increases data integrity. When data digest is enabled, the system performs a checksum over each PDU's data part and verifies using the CRC32C algorithm. **NOTE** Systems that use Intel Nehalem processors offload the iSCSI digest calculations for software iSCSI, thus reducing the impact on performance.	Software iSCSI Dependent Hardware iSCSI
Maximum Outstanding R2T	Defines the R2T (Ready to Transfer) PDUs that can be in transition before an acknowledge PDU is received.	Software iSCSI Dependent Hardware iSCSI
First Burst Length	Specifies the maximum amount of unsolicited data an iSCSI initiator can send to the target during the execution of a single SCSI command, in bytes.	Software iSCSI Dependent Hardware iSCSI
Maximum Burst Length	Maximum SCSI data payload in a Data-In or a solicited Data-Out iSCSI sequence, in bytes.	Software iSCSI Dependent Hardware iSCSI
Maximum Receive Data Segment Length	Maximum data segment length, in bytes, that can be received in an iSCSI PDU.	Software iSCSI Dependent Hardware iSCSI
Session Recovery Timeout	Specifies the amount of time, in seconds, that can lapse while a session recovery is performed. If the timeout exceeds its limit, the iSCSI initiator terminates the session.	Software iSCSI Dependent Hardware iSCSI
No-Op Interval	Specifies the time interval, in seconds, between NOP-Out requests sent from your iSCSI initiator to an iSCSI target. The NOP-Out requests serve as the ping mechanism to verify that a connection between the iSCSI initiator and the iSCSI target is active.	Software iSCSI Dependent Hardware iSCSI

Table 9-5. Additional Parameters for iSCSI Initiators (Continued)

Advanced Parameter	Description	Configurable On
No-Op Timeout	Specifies the amount of time, in seconds, that can lapse before your host receives a NOP-In message. The message is sent by the iSCSI target in response to the NOP-Out request. When the no-op timeout limit is exceeded, the initiator terminates the current session and starts a new one.	Software iSCSI Dependent Hardware iSCSI
ARP Redirect	Allows storage systems to move iSCSI traffic dynamically from one port to another. ARP is required by storage systems that do array-based failover.	Software iSCSI Dependent Hardware iSCSI Independent Hardware iSCSI
Delayed ACK	Allows systems to delay acknowledgment of received data packets.	Software iSCSI Dependent Hardware iSCSI

Configure Advanced Parameters for iSCSI

The advanced iSCSI settings control such parameters as header and data digest, ARP redirection, delayed ACK, and so on. Generally, you do not need to change these settings because your host works with the assigned predefined values.

CAUTION Do not make any changes to the advanced iSCSI settings unless you are working with the VMware support team or otherwise have thorough information about the values to provide for the settings.

Required privilege: **Host.Configuration.Storage Partition Configuration**

Procedure

1 Access the iSCSI Initiator Properties dialog box.

2 To configure advanced parameters at the initiator level, on the General tab, click **Advanced**. Proceed to Step 4.

3 Configure advanced parameters at the target level.

At the target level, advanced parameters can be configured only for software and dependent hardware iSCSI adapters.

a Select either the **Dynamic Discovery** tab or **Static Discovery** tab.

b From the list of available targets, select a target to configure and click **Settings > Advanced**.

4 Enter any required values for the advanced parameters you want to modify and click **OK** to save your changes.

iSCSI Session Management

To communicate with each other, iSCSI initiators and targets establish iSCSI sessions. You can review and manage iSCSI sessions using vSphere CLI.

By default, software iSCSI and dependent hardware iSCSI initiators start one iSCSI session between each initiator port and each target port. If your iSCSI initiator or target have more than one port, your host can have multiple sessions established. The default number of sessions for each target equals the number of ports on the iSCSI adapter times the number of target ports.

Using vSphere CLI, you can display all current sessions to analyze and debug them. To create more paths to storage systems, you can increase the default number of sessions by duplicating existing sessions between the iSCSI adapter and target ports.

You can also establish a session to a specific target port. This can be useful if your host connects to a single-port storage system that, by default, presents only one target port to your initiator, but can redirect additional sessions to a different target port. Establishing a new session between your iSCSI initiator and another target port creates an additional path to the storage system.

The following considerations apply to iSCSI session management:

- Some storage systems do not support multiple sessions from the same initiator name or endpoint. Attempts to create multiple sessions to such targets can result in unpredictable behavior of your iSCSI environment.
- Storage vendors can provide automatic session managers. Using the automatic session manages to add or delete sessions, does not guarantee lasting results and can interfere with the storage performance.

Review iSCSI Sessions

Use the vCLI command to display iSCSI sessions between an iSCSI adapter and a storage system.

In the procedure, **--server=*server_name*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

- To list iSCSI sessions, run the following command:

  esxcli --server=*server_name* iscsi session list

  The command takes these options:

Option	Description
-A|--adapter=*str*	The iSCSI adapter name, for example, vmhba34.
-s|--isid=*str*	The iSCSI session identifier.
-n|--name=*str*	The iSCSI target name, for example, iqn.X.

Add iSCSI Sessions

Use the vCLI to add an iSCSI session for a target you specify or to duplicate an existing session. By duplicating sessions, you increase the default number of sessions and create additional paths to storage systems.

In the procedure, **--server=*server_name*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

- To add or duplicate an iSCSI session, run the followign command:

  esxcli --server=*server_name* iscsi session add

  The command takes these options:

Option	Description
-A|--adapter=*str*	The iSCSI adapter name, for example, vmhba34. This option is required.
-s|--isid=*str*	The ISID of a session to duplicate. You can find it by listing all session.
-n|--name=*str*	The iSCSI target name, for example, iqn.X.

What to do next

Rescan the iSCSI adapter.

Remove iSCSI Sessions

Use the vCLI command to remove an iSCSI session between an iSCSI adapter and a target.

In the procedure, **--server=*server_name*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

- To remove a session, run the following command:

  esxcli --server=*server_name* iscsi session remove

  The command takes these options:

Option	Description
-A|--adapter=*str*	The iSCSI adapter name, for example, vmhba34. This option is required.
-s|--isid=*str*	The ISID of a session to remove. You can find it by listing all session.
-n|--name=*str*	The iSCSI target name, for example, iqn.X.

What to do next

Rescan the iSCSI adapter.

Modifying iSCSI Storage Systems for ESXi 10

After you configure your iSCSI initiators and storage, you might need to modify your storage system to ensure that it works properly with your ESXi implementation.

This section discusses many of the iSCSI storage systems supported in conjunction with VMware ESXi. For each device, it lists major known potential issues, points to vendor-specific information (if available), or includes information from VMware knowledge base articles.

NOTE Information in this section is updated only with each release. New information might already be available. Also, other iSCSI storage systems are supported but are not covered in this chapter. Consult the most recent *Storage/SAN Compatibility*, check with your storage vendor, and explore the VMware knowledge base articles.

This chapter includes the following topics:

- "Testing ESXi iSCSI SAN Configurations," on page 91
- "General Considerations for iSCSI SAN Storage Systems," on page 92
- "EMC CLARiiON Storage Systems," on page 92
- "EMC Symmetrix Storage Systems," on page 93
- "Enable HP StorageWorks MSA1510i to Communicate with ESXi," on page 93
- "HP StorageWorks EVA Storage Systems," on page 94
- "NetApp Storage Systems," on page 95
- "Dell EqualLogic Storage Systems," on page 95
- "HP StorageWorks SAN/iQ Storage Systems," on page 95
- "Dell PowerVault MD3000i Storage Systems," on page 96
- "iSCSI Targets in vApps," on page 96

Testing ESXi iSCSI SAN Configurations

ESXi supports a variety of SAN storage systems in different configurations. Generally, VMware tests ESXi with supported storage systems for basic connectivity, HBA failover, and so on.

Not all storage devices are certified for all features and capabilities of ESXi, and vendors might have specific positions of support with regard to ESXi.

VMware tests ESXi with storage systems in the following configurations:

Basic Connectivity	Tests whether ESXi can recognize and operate with the storage system. This configuration does not allow for multipathing or any type of failover.
iSCSI Failover	The server is equipped with multiple iSCSI HBAs or NICs. The server is robust to HBA or NIC failure.
Storage Port Failover	The server is attached to multiple storage ports and is robust to storage port failures and switch failures.
Booting from a SAN	The host successfully boots from a LUN configured on the iSCSI SAN.

General Considerations for iSCSI SAN Storage Systems

When you prepare your iSCSI SAN storage system to work with ESXi, you need to follow specific general requirements that apply to all storage systems.

For all storage systems, the following general requirements exist:

- LUNs must be presented to each HBA of each host with the same LUN ID number. If different numbers are used, the ESXi hosts do not recognize different paths to the same LUN. Because instructions on how to configure identical SAN LUN IDs are vendor-specific, consult your storage documentation for more information.
- Unless specified for individual storage systems discussed in this chapter, set the host type for LUNs presented to ESXi to `Linux` or `Linux Cluster`, if applicable to your storage system. The method ESXi uses to access the storage system is most compatible with Linux access, however, this can vary depending on the storage system you are using.
- If you are using vMotion, DRS, or HA, make sure that source and target hosts for virtual machines can see the same LUNs with identical LUN IDs. SAN administrators might find it counterintuitive to have multiple hosts see the same LUNs because they might be concerned about data corruption. However, VMFS prevents multiple virtual machines from writing to the same file at the same time, so provisioning the LUNs to all required ESXi system is appropriate.
- If you do not have CHAP authentication set up on the LUNs that are being accessed, you must also disable CHAP on the ESXi host. Otherwise, authentication of the storage system fails, although the LUNs have no CHAP requirement.

EMC CLARiiON Storage Systems

EMC CLARiiON storage systems work with ESXi hosts in iSCSI SAN configurations. Generally, you use the EMC software to perform configurations.

This is an active-passive disk array, so any related issues that apply to all active-passive disk arrays are relevant. In addition, keep in mind the following:

- To avoid the possibility of path thrashing, the default multipathing policy is Most Recently Used, not Fixed. The ESXi system sets the default policy when it identifies the storage system.
- To boot from a SAN, choose the active storage processor for the boot LUN's target in the HBA BIOS.
- Port binding support on EMC CLARiiON storage systems requires initiators in different subnets if the storage is using a version of FLARE earlier than FLARE 30. See vendor documentation for additional details.
- For ESXi to support EMC CLARiiON with ALUA, check the HCLs to make sure that you use the correct firmware version on the storage array. For additional information, contact your storage vendor.

EMC Symmetrix Storage Systems

To work with ESXi, EMC Symmetrix storage systems require certain specific settings. Use EMC software to configure the storage system.

The following settings are required for ESXi operations on the Symmetrix networked storage system:

- Common serial number (C)
- Auto negotiation (EAN) enabled
- SCSI 3 (SC3) set (enabled)
- Unique world wide name (UWN)
- SPC-2 (Decal) (SPC2) SPC-2 flag is required

NOTE The ESXi host considers any LUNs from a Symmetrix storage system that have a capacity of 50MB or less as management LUNs. These LUNs are also known as pseudo or gatekeeper LUNs. These LUNs appear in the EMC Symmetrix Management Interface and should not be used to hold data.

Enable HP StorageWorks MSA1510i to Communicate with ESXi

This section describes the setup and configuration steps needed to allow an HP StorageWorks MSA1510i storage system to communicate with ESXi hosts.

Procedure

1 Install, connect, and power up the network devices as detailed in the vendor installation document.

2 Obtain the IP address assigned to the MSA1510i controller management port.

   a Scroll through the messages on the LCD panel until the following message appears: `603 Port MA0 IP` *`address`*

   b Record the management port IP address that appears in **Basic MSA1510i information**.

3 From the server or a workstation on the MSA1510i LAN segment, open a Web browser and enter the address obtained in Step 2.

4 When prompted, enter the default access permissions.

   - User name: `root`
   - Password: `root`

5 When prompted, set a unique user name and password.

6 Using the wizard, complete the following actions.

Option	Description
Storage configuration	a Set the Fault Tolerant mode (RAID mode). b Assign a spare disk for appropriate RAID level.
iSCSI configuration (configure an iSCSI portal)	a Select a data port. b Assign an IP address to the data port. c VLANs are set up on the switch and are used as one method of controlling access to the storage. If you are using VLANs, enter the VLAN ID to use (0 = not used). d The wizard suggests a default iSCSI Target Name and iSCSI Target Alias. Accept the default or enter user-defined values. **NOTE** To configure the remaining data ports, complete the Initial System Configuration Wizard process, and then use tasks available on the **Configure** tab.
Login settings	Enter login settings.
Management settings	Enter management settings.

7 Click **Finish** to apply the configuration settings.

NOTE Wizards are available for basic configuration tasks only. Use the **Manage** and **Configure** tabs to view and change your configuration.

What to do next

After initial setup, perform the following tasks to complete the configuration:

- Create an array.
- Create a logical drive.
- Create a target.
- Create a portal group.
- Associate or assign the portals created using the wizard with the portal group created.
- Map logical drives to the target.
- Add initiators (initiator IQN name and alias).
- Update the ACLs of the logical drives to provide access to initiators (select the list of initiators to access the logical drive).

HP StorageWorks EVA Storage Systems

The two types of HP StorageWorks EVA systems are EVA_GL, an active-passive system, and EVA_XL, an active-active system. For the systems to work with ESXi, certain specific settings are required.

Set the connection type to **Custom** when you present a LUN to an ESXi host. The value is one of the following:

- For HP EVAgl 3000/5000 (active-passive), use the `000000002200282E` host mode type.
- For HP EVAgl firmware 4.001 (active-active firmware for GL series) and above, use the `VMware` host mode type.
- For EVA4000/6000/8000 active-active arrays with firmware earlier than 5.031, use the `000000202200083E` host mode type.
- For EVA4000/6000/8000 active-active arrays with firmware 5.031 and later, use the `VMware` host mode type.

Otherwise, EVA systems do not require special configuration changes to work with an ESXi system.

NetApp Storage Systems

For NetApp storage systems to communicate within an ESXi environment, you must perform specific configuration steps.

For additional documentation on NetApp and VMware best practices and SAN solutions, search the NetApp web page.

Table 10-1. Configuration Steps

Configuration Step	Description
Disable ALUA.	If any of your iSCSI initiators are a part of an initiator group (igroup), disable ALUA on the NetApp filer.
Set up multipathing.	When you set up multipathing between two iSCSI HBAs and multiple ports on a NetApp storage system, give each HBA a different iSCSI initiator name. The NetApp storage system only permits one connection for each target and each initiator. Attempts to make additional connections cause the first connection to drop. Therefore, a single HBA should not attempt to connect to multiple IP addresses associated with the same NetApp target.
Set LUN type and initiator group type.	Set the appropriate LUN type and initiator group type for the storage system: ■ LUN type – VMware (if VMware type is not available, use Linux). ■ Initiator group type – VMware (if VMware type is not available, use Linux).
Provision storage.	Use either FilerView or CLI.

Dell EqualLogic Storage Systems

When setting up your EqualLogic storage systems to work in an ESXi implementation, you must address certain specific issues.

Follow these requirements:

- Multipathing. No special setup is needed because EqualLogic storage systems support storage-processor failover that is transparent to iSCSI. Multiple iSCSI HBAs or NICs can connect to the same target or LUN on the storage side. EqualLogic provides a multipathing extension module that can be installed on ESXi to provide additional capabilities.

  For information about the custom multipathing configuration packages, contact Dell EqualLogic.

- Creating iSCSI LUNs. From the EqualLogic web portal, right-click **Volumes**, and then select **Create Volume**.
- Enable ARP redirection for ESXi hardware iSCSI HBAs.
- EqualLogic storage systems impose a maximum limit of 1024 iSCSI connections per storage pool and 2048 connections per storage group.

For more information about configuring and using EqualLogic storage systems, see the vendor's documentation.

HP StorageWorks SAN/iQ Storage Systems

HP StorageWorks (formerly LeftHand) SAN/iQ systems support ESXi iSCSI connections from a software initiator and hardware initiator.

When configuring SAN/iQ, enable automatic volume resignaturing for SAN/iQ storage devices to allow access to SAN/iQ snapshots and remote copies.

For more information on configuring HP StorageWorks storage for VMware vSphere, see the vendor documentation related to VMware.

Basic configuration steps include several tasks.

1 Install SAN/iQ storage nodes.

2 Create SAN/iQ management groups and clusters.

3 Create volumes.

4 Assign volumes to authentication groups and volume lists.

5 Enable ARP redirection on hardware iSCSI HBAs.

As a best practice, configure virtual IP load balancing in SAN/iQ for all ESXi authentication groups.

Dell PowerVault MD3000i Storage Systems

When you configure mutual CHAP for the MD3000i iSCSI storage systems, special considerations that apply.

When you configure mutual CHAP for the MD3000i iSCSI array, follow these guidelines:

- On the MD3000i storage system, mutual CHAP configuration requires only a CHAP secret.
- On the ESXi host, mutual CHAP configuration requires both the name and CHAP secret. When configuring mutual CHAP on the ESXi host, enter the IQN name of the target as the mutual CHAP name. Make sure the CHAP secret matches the one set on the array.

iSCSI Targets in vApps

If you use an iSCSI target in a virtual appliance, for example HP LeftHand P4000 VSA, the host should connect to the target through the software iSCSI adapter rather than a hardware iSCSI adapter.

11 Booting from iSCSI SAN

When you set up your host to boot from a SAN, your host's boot image is stored on one or more LUNs in the SAN storage system. When the host starts, it boots from the LUN on the SAN rather than from its local disk.

You can use boot from the SAN if you do not want to handle maintenance of local storage or have diskless hardware configurations, such as blade systems.

ESXi supports different methods of booting from the iSCSI SAN.

Table 11-1. Boot from iSCSI SAN support

Independent Hardware iSCSI	Software iSCSI and Dependent Hardware iSCSI
Configure the iSCSI HBA to boot from the SAN. For information on configuring the HBA, see "Configure Independent Hardware iSCSI Adapter for SAN Boot," on page 98	Use the network adapter that supports the iBFT. For information, see "iBFT iSCSI Boot Overview," on page 99.

This chapter includes the following topics:

- "General Boot from iSCSI SAN Recommendations," on page 97
- "Prepare the iSCSI SAN," on page 98
- "Configure Independent Hardware iSCSI Adapter for SAN Boot," on page 98
- "iBFT iSCSI Boot Overview," on page 99

General Boot from iSCSI SAN Recommendations

If you plan to set up and use an iSCSI LUN as the boot device for your host, you need to follow certain general guidelines.

The following guidelines apply to booting from independent hardware iSCSI and iBFT.

- Review any vendor recommendations for the hardware you use in your boot configuration.
- For installation prerequisites and requirements, review *vSphere Installation and Setup*.
- Use static IP addresses to reduce the chances of DHCP conflicts.
- Use different LUNs for VMFS datastores and boot partitions.
- Configure proper ACLs on your storage system.
  - The boot LUN should be visible only to the host that uses the LUN. No other host on the SAN should be permitted to see that boot LUN.
  - If a LUN is used for a VMFS datastore, it can be shared by multiple hosts. ACLs on the storage systems can allow you to do this.

- Configure a diagnostic partition.
  - With independent hardware iSCSI only, you can place the diagnostic partition on the boot LUN. If you configure the diagnostic partition in the boot LUN, this LUN cannot be shared across multiple hosts. If a separate LUN is used for the diagnostic partition, it can be shared by multiple hosts.
  - If you boot from SAN using iBFT, you cannot set up a diagnostic partition on a SAN LUN. Instead, you use the vSphere Management Assistant (vMA) to collect diagnostic information from your host and store it for analysis.

    See "Collecting Diagnostic Information," on page 104.

Prepare the iSCSI SAN

Before you configure your host to boot from an iSCSI LUN, prepare and configure your storage area network.

CAUTION If you use scripted installation to install ESXi when booting from a SAN, you must take special steps to avoid unintended data loss.

Procedure

1 Connect network cables, referring to any cabling guide that applies to your setup.

2 Ensure IP connectivity between your storage system and server.

  This includes proper configuration of any routers or switches on your storage network. Storage systems must be able to ping the iSCSI adapters in your hosts.

3 Configure the storage system.

  a Create a volume (or LUN) on the storage system for your host to boot from.

  b Configure the storage system so that your host has access to the assigned LUN.

    This could involve updating ACLs with the IP addresses, iSCSI names, and the CHAP authentication parameter you use on your host. On some storage systems, in addition to providing access information for the ESXi host, you must also explicitly associate the assigned LUN with the host.

  c Ensure that the LUN is presented to the host correctly.

  d Ensure that no other system has access to the configured LUN.

  e Record the iSCSI name and IP addresses of the targets assigned to the host.

    You must have this information to configure your iSCSI adapters.

Configure Independent Hardware iSCSI Adapter for SAN Boot

If your ESXi host uses an independent hardware iSCSI adapter, such as QLogic HBA, you need to configure the adapter to boot from the SAN.

This procedure discusses how to enable the QLogic iSCSI HBA to boot from the SAN. For more information and more up-to-date details about QLogic adapter configuration settings, see the QLogic web site.

Prerequisites

Because you first need to boot from the VMware installation media, set up your host to boot from CD/DVD-ROM. To achieve this, change the system boot sequence in your system BIOS setup.

Procedure

1 Insert the installation CD/DVD in the CD/DVD-ROM drive and reboot the host.

2 Use the BIOS to set the host to boot from the CD/DVD-ROM drive first.

3 During server POST, press Crtl+q to enter the QLogic iSCSI HBA configuration menu.

4 Select the I/O port to configure.

By default, the Adapter Boot mode is set to Disable.

5 Configure the HBA.

a From the **Fast!UTIL Options** menu, select **Configuration Settings > Host Adapter Settings**.

b Configure the following settings for your host adapter: initiator IP address, subnet mask, gateway, initiator iSCSI name, and CHAP (if required).

6 Configure iSCSI settings.

See "Configure iSCSI Boot Settings," on page 99.

7 Save your changes and restart the system.

Configure iSCSI Boot Settings

When setting up your ESXi host to boot from iSCSI, you need to configure iSCSI boot settings.

Procedure

1 From the **Fast!UTIL Options** menu, select **Configuration Settings > iSCSI Boot Settings**.

2 Before you can set SendTargets, set Adapter Boot mode to **Manual**.

3 Select **Primary Boot Device Settings**.

a Enter the discovery **Target IP** and **Target Port**.

b You can leave the **Boot LUN** and **iSCSI Name** fields blank if only one iSCSI target and one LUN are at the specified address to boot from. Otherwise, you must specify these fields to ensure that you do not boot from a volume for some other system. After the target storage system is reached, these fields will be populated after a rescan.

c Save changes.

4 From the **iSCSI Boot Settings** menu, select the primary boot device. An auto rescan of the HBA is made to find new target LUNS.

5 Select the iSCSI target.

NOTE If more then one LUN exists within the target, you can choose a specific LUN ID by pressing **Enter** after you locate the iSCSI device.

---

6 Return to the **Primary Boot Device Setting** menu. After the rescan, the **Boot LUN**and **iSCSI Name** fields are populated. Change the value of **Boot LUN** to the desired LUN ID.

iBFT iSCSI Boot Overview

ESXi hosts can boot from an iSCSI SAN using the software or dependent hardware iSCSI adapters and network adapters.

To deploy ESXi and boot from the iSCSI SAN, the host must have an iSCSI boot capable network adapter that supports the iSCSI Boot Firmware Table (iBFT) format. The iBFT is a method of communicating parameters about the iSCSI boot device to an operating system.

Before installing ESXi and booting from the iSCSI SAN, configure the networking and iSCSI boot parameters on the network adapter and enable the adapter for the iSCSI boot. Because configuring the network adapter is vendor specific, review your vendor documentation for instructions.

When you first boot from iSCSI, the iSCSI boot firmware on your system connects to an iSCSI target. If login is successful, the firmware saves the networking and iSCSI boot parameters in the iBFT and stores the table in the system's memory. The system uses this table to configure its own iSCSI connection and networking and to start up.

The following list describes the iBFT iSCSI boot sequence.

1 When restarted, the system BIOS detects the iSCSI boot firmware on the network adapter.

2 The iSCSI boot firmware uses the preconfigured boot parameters to connect with the specified iSCSI target.

3 If the connection to the iSCSI target is successful, the iSCSI boot firmware writes the networking and iSCSI boot parameters in to the iBFT and stores the table in the system memory.

NOTE The system uses this table to configure its own iSCSI connection and networking and to start up.

4 The BIOS boots the boot device.

5 The VMkernel starts loading and takes over the boot operation.

6 Using the boot parameters from the iBFT, the VMkernel connects to the iSCSI target.

7 After the iSCSI connection is established, the system boots.

iBFT iSCSI Boot Considerations

When you boot the ESXi host from iSCSI using iBFT-enabled network adapters, certain considerations apply.

The iBFT iSCSI boot does not support the following items:

- IPv6
- Failover for the iBFT-enabled network adapters

NOTE Update your NIC's boot code and iBFT firmware using vendor supplied tools before trying to install and boot VMware ESXi. Consult vendor documentation and VMware HCL for supported boot code and iBFT firmware versions for VMware ESXi iBFT boot. The boot code and iBFT firmware released by vendors prior to the ESXi 4.1 release might not work.

After you set up your host to boot from iBFT iSCSI, the following restrictions apply:

- You cannot disable the software iSCSI adapter. If the iBFT configuration is present in the BIOS, the host re-enables the software iSCSI adapter during each reboot.

  NOTE If you do not use the iBFT-enabled network adapter for the iSCSI boot and do not want the software iSCSI adapter to be always enabled, remove the iBFT configuration from the network adapter.

- You cannot remove the iBFT iSCSI boot target using the vSphere Client. The target appears on the list of adapter static targets.

Configuring iBFT Boot from SAN

You can boot from the iSCSI SAN using the software iSCSI adapter or a dependent hardware iSCSI adapter and a network adapter. The network adapter must support iBFT.

When you set up your host to boot with iBFT, you perform a number of tasks.

1 Configure iSCSI Boot Parameters on page 101

To begin an iSCSI boot process, a network adapter on your host must have a specially configured iSCSI boot firmware. When you configure the firmware, you specify the networking and iSCSI parameters and enable the adapter for the iSCSI boot.

2 Change Boot Sequence in BIOS on page 101

When setting up your host to boot from iBFT iSCSI, change the boot sequence to force your host to boot in an appropriate order.

3 Install ESXi to iSCSI Target on page 102

When setting up your host to boot from iBFT iSCSI, install the ESXi image to the target LUN.

4 Boot ESXi from iSCSI Target on page 102

After preparing the host for an iBFT iSCSI boot and copying the ESXi image to the iSCSI target, perform the actual boot.

Configure iSCSI Boot Parameters

To begin an iSCSI boot process, a network adapter on your host must have a specially configured iSCSI boot firmware. When you configure the firmware, you specify the networking and iSCSI parameters and enable the adapter for the iSCSI boot.

Configuration on the network adapter can be dynamic or static. If you use the dynamic configuration, you indicate that all target and initiator boot parameters are acquired using DHCP. For the static configuration, you manually enter data that includes your host's IP address and initiator IQN, and the target parameters.

Procedure

- On the network adapter that you use for the boot from iSCSI, specify networking and iSCSI parameters.

  Because configuring the network adapter is vendor specific, review your vendor documentation for instructions.

Change Boot Sequence in BIOS

When setting up your host to boot from iBFT iSCSI, change the boot sequence to force your host to boot in an appropriate order.

Change the BIOS boot sequence to the following sequence:

- iSCSI
- DVD-ROM

Because changing the boot sequence in the BIOS is vendor specific, refer to vendor documentation for instructions. The following sample procedure explains how to change the boot sequence on a Dell host with a Broadcom network adapter.

Procedure

1 Turn on the host.

2 During Power-On Self-Test (POST), press F2 to enter the BIOS Setup.

3 In the BIOS Setup, select **Boot Sequence** and press Enter.

4 In the Boot Sequence menu, arrange the bootable items so that iSCSI precedes the DVD-ROM.

5 Press Esc to exit the Boot Sequence menu.

6 Press Esc to exit the BIOS Setup.

7 Select **Save Changes** and click **Exit** to exit the BIOS Setup menu.

Install ESXi to iSCSI Target

When setting up your host to boot from iBFT iSCSI, install the ESXi image to the target LUN.

Prerequisites

- Configure iSCSI boot firmware on your boot NIC to point to the target LUN that you want to use as the boot LUN.
- Change the boot sequence in the BIOS so that iSCSI precedes the DVD-ROM.
- If you use Broadcom adapters, set **Boot to iSCSI target** to **Disabled**.

Procedure

1 Insert the installation media in the CD/DVD-ROM drive and restart the host.

2 When the installer starts, follow the typical installation procedure.

3 When prompted, select the iSCSI LUN as the installation target.

The installer copies the ESXi boot image to the iSCSI LUN.

4 After the system restarts, remove the installation DVD.

Boot ESXi from iSCSI Target

After preparing the host for an iBFT iSCSI boot and copying the ESXi image to the iSCSI target, perform the actual boot.

Prerequisites

- Configure the iSCSI boot firmware on your boot NIC to point to the boot LUN.
- Change the boot sequence in the BIOS so that iSCSI precedes the boot device.
- If you use Broadcom adapters, set **Boot to iSCSI target** to **Enabled**

Procedure

1 Restart the host.

The host boots from the iSCSI LUN using iBFT data. During the first boot, the iSCSI initialization script sets up default networking. The network setup is persistent after subsequent reboots.

2 (Optional) Adjust networking configuration using the vSphere Client.

Networking Best Practices

To boot the ESXi host from iSCSI using iBFT, you must properly configure networking.

To achieve greater security and better performance, have redundant network adapters on the host.

How you set up all the network adapters depends on whether your environment uses shared or isolated networks for the iSCSI traffic and host management traffic.

Shared iSCSI and Management Networks

Configure the networking and iSCSI parameters on the first network adapter on the host. After the host boots, you can add secondary network adapters to the default port group.

Isolated iSCSI and Management Networks

When you configure isolated iSCSI and management networks, follow these guidelines to avoid bandwidth problems.

- Your isolated networks must be on different subnets.
- If you use VLANs to isolate the networks, they must have different subnets to ensure that routing tables are properly set up.
- VMware recommends that you configure the iSCSI adapter and target to be on the same subnet. If you set up the iSCSI adapter and target on different subnets, the following restrictions apply:
  - The default VMkernel gateway must be able to route both the management and iSCSI traffic.
  - After you boot your host, you can use the iBFT-enabled network adapter only for iBFT. You cannot use the adapter for other iSCSI traffic.
- Use the first physical network adapter for the management network.
- Use the second physical network adapter for the iSCSI network. Make sure to configure the iBFT.
- After the host boots, you can add secondary network adapters to both the management and iSCSI networks.

Change iBFT iSCSI Boot Settings

If settings, such as the IQN name, IP address, and so on, change on the iSCSI storage or your host, update the iBFT. This task assumes that the boot LUN and the data stored on the LUN remain intact.

Procedure

1 Shut down the ESXi host.

2 Change iSCSI storage settings.

3 Update the iBFT on the host with the new settings.

4 Restart the host.

The host boots using the new information stored in the iBFT.

Troubleshooting iBFT iSCSI Boot

The topics in this section help you to identify and solve problems you might encounter when using iBFT iSCSI boot.

Loss of System's Gateway Causes Loss of Network Connectivity

You lose network connectivity when you delete a port group associated with the iBFT network adapter.

Problem

A loss of network connectivity occurs after you delete a port group.

Cause

When you specify a gateway in the iBFT-enabled network adapter during ESXi installation, this gateway becomes the system's default gateway. If you delete the port group associated with the network adapter, the system's default gateway is lost. This action causes the loss of network connectivity.

Solution

Do not set an iBFT gateway unless it is required. If the gateway is required, after installation, manually set the system's default gateway to the one that the management network uses.

Changing iSCSI Boot Parameters Causes ESXi to Boot in Stateless Mode

Changing iSCSI boot parameters on the network adapter after the first boot does not update the iSCSI and networking configuration on the ESXi host.

Problem

If you change the iSCSI boot parameters on the network adapter after the first ESXi boot from iSCSI, the host will boot in a stateless mode.

Cause

The firmware uses the updated boot configuration and is able to connect to the iSCSI target and load the ESXi image. However, when loaded, the system does not pick up the new parameters, but continues to use persistent networking and iSCSI parameters from the previous boot. As a result, the host cannot connect to the target and boots in the stateless mode.

Solution

1 Use the vSphere Client to connect to the ESXi host.

2 Re-configure the iSCSI and networking on the host to match the iBFT parameters.

3 Perform a rescan.

Collecting Diagnostic Information

If you boot from SAN using iBFT and your ESXi host does not have a local diagnostic partition, you need to set up the vSphere Management Assistant (vMA) to collect diagnostic information from your host and store it for analysis.

If your ESXi host experiences serious problems, it creates a file with diagnostic information that can be later analyzed to determine the cause of the problems.

You set up your host to act as a net dump client that transfers the diagnostic file over the network. vMA acts as the net dump server that collects the diagnostic information and saves it to `/var/core` on the server.

When you set up the net dump environment, follow these guidelines:

- Do not install vMA on the same physical host where you set up the net dump client.
- If you have multiple ESXi hosts that require the net dump configuration, configure each host separately. One vMA instance is sufficient for collecting the core dump files from multiple hosts.

Configure vSphere Management Assistant

The vSphere Management Assistant (vMA) is a virtual machine that you configure to act as a net dump server that collects diagnostic information from your ESXi host.

Prerequisites

Deploy and configure vMA on a physical server different form the server where you set up the net dump client. For information, see the *vSphere Management Assistant Guide*.

Procedure

1 Log in to vMA as administrator.

2 Enable the net dump server on vMA.

```
# sudo chkconfig vmware-netdumper on
```

3 Start the net dump server.

```
# sudo/etc/init.d/vmware-netdumper start

# sudo lokkit --quiet --port=6500:udp
```

Configure ESXi Host

Use vSphere CLI to configure the ESXi host to act as the net dump client that transfers diagnostic information over the network.

In the procedure, **--server=*server_name*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Configure vMA as the net dump server and obtain its IP address.

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

1 Enter the IP address of the ESXi host.

```
# esxcli --server=server_name system settings advanced -s IP_address /Net/NetdumpVmkIP
```

2 Specify the network adapter to use.

```
# esxcli --server=server_name system settings advanced -s vmnic /Net/NetdumpVmkNic
```

3 Specify the port group attached to the network adapter.

```
# esxcli --server=server_name system settings advanced -s portgroup /Net/NetdumpVmkPG
```

4 Specify the IP address of the net dump server.

```
# esxcli --server=server_name system settings advanced -
s IP_address_netdump /Net/NetdumpServerIP
```

5 (Optional) Enter the IP address of gateway to reach the net dump server.

```
# esxcli --server=server_name system settings advanced -
s IP_address_gateway /Net/NetdumpServerGateway
```

12 Best Practices for iSCSI Storage

When using ESXi with the iSCSI SAN, follow best practices that VMware offers to avoid problems.

Check with your storage representative if your storage system supports Storage API - Array Integration hardware acceleration features. If it does, refer to your vendor documentation for information on how to enable hardware acceleration support on the storage system side. For more information, see Chapter 18, "Storage Hardware Acceleration," on page 173.

This chapter includes the following topics:

- "Preventing iSCSI SAN Problems," on page 107
- "Optimizing iSCSI SAN Storage Performance," on page 108
- "Checking Ethernet Switch Statistics," on page 111
- "iSCSI SAN Configuration Checklist," on page 111

Preventing iSCSI SAN Problems

When using ESXi in conjunction with a SAN, you must follow specific guidelines to avoid SAN problems.

You should observe these tips for avoiding problems with your SAN configuration:

- Place only one VMFS datastore on each LUN. Multiple VMFS datastores on one LUN is not recommended.
- Do not change the path policy the system sets for you unless you understand the implications of making such a change.
- Document everything. Include information about configuration, access control, storage, switch, server and iSCSI HBA configuration, software and firmware versions, and storage cable plan.
- Plan for failure:
  - Make several copies of your topology maps. For each element, consider what happens to your SAN if the element fails.
  - Cross off different links, switches, HBAs and other elements to ensure you did not miss a critical failure point in your design.
- Ensure that the iSCSI HBAs are installed in the correct slots in the ESXi host, based on slot and bus speed. Balance PCI bus load among the available busses in the server.
- Become familiar with the various monitor points in your storage network, at all visibility points, including ESXi performance charts, Ethernet switch statistics, and storage performance statistics.
- Be cautious when changing IDs of the LUNs that have VMFS datastores being used by your host. If you change the ID, virtual machines running on the VMFS datastore will fail.

If there are no running virtual machines on the VMFS datastore, after you change the ID of the LUN, you must use rescan to reset the ID on your host. For information on using rescan, see "Perform Storage Rescan," on page 124.

- If you need to change the default iSCSI name of your iSCSI adapter, make sure the name you enter is worldwide unique and properly formatted. To avoid storage access problems, never assign the same iSCSI name to different adapters, even on different hosts.

Optimizing iSCSI SAN Storage Performance

Several factors contribute to optimizing a typical SAN environment.

If the network environment is properly configured, the iSCSI components provide adequate throughput and low enough latency for iSCSI initiators and targets. If the network is congested and links, switches or routers are saturated, iSCSI performance suffers and might not be adequate for ESXi environments.

Storage System Performance

Storage system performance is one of the major factors contributing to the performance of the entire iSCSI environment.

If issues occur with storage system performance, consult your storage system vendor's documentation for any relevant information.

When you assign LUNs, remember that you can access each shared LUN through a number of hosts, and that a number of virtual machines can run on each host. One LUN used by the ESXi host can service I/O from many different applications running on different operating systems. Because of this diverse workload, the RAID group that contains the ESXi LUNs should not include LUNs that other hosts use that are not running ESXi for I/O intensive applications.

Enable read caching and write caching.

Load balancing is the process of spreading server I/O requests across all available SPs and their associated host server paths. The goal is to optimize performance in terms of throughput (I/O per second, megabytes per second, or response times).

SAN storage systems require continual redesign and tuning to ensure that I/O is load balanced across all storage system paths. To meet this requirement, distribute the paths to the LUNs among all the SPs to provide optimal load balancing. Close monitoring indicates when it is necessary to manually rebalance the LUN distribution.

Tuning statically balanced storage systems is a matter of monitoring the specific performance statistics (such as I/O operations per second, blocks per second, and response time) and distributing the LUN workload to spread the workload across all the SPs.

Server Performance with iSCSI

You must consider several factors to ensure optimal server performance.

Each server application must have access to its designated storage with the following conditions:

- High I/O rate (number of I/O operations per second)
- High throughput (megabytes per second)
- Minimal latency (response times)

Because each application has different requirements, you can meet these goals by choosing an appropriate RAID group on the storage system. To achieve performance goals, perform the following tasks:

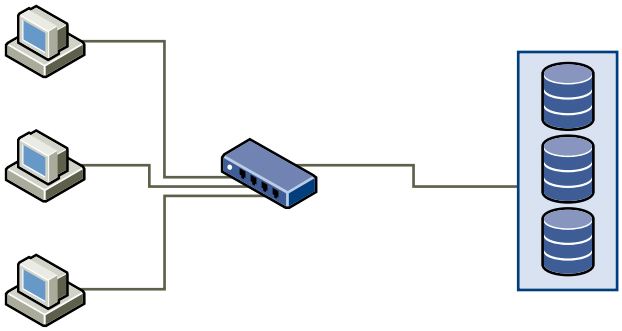
- Place each LUN on a RAID group that provides the necessary performance levels. Pay attention to the activities and resource utilization of other LUNS in the assigned RAID group. A high-performance RAID group that has too many applications doing I/O to it might not meet performance goals required by an application running on the ESXi host.
- Provide each server with a sufficient number of network adapters or iSCSI hardware adapters to allow maximum throughput for all the applications hosted on the server for the peak period. I/O spread across multiple ports provides higher throughput and less latency for each application.
- To provide redundancy for software iSCSI, make sure the initiator is connected to all network adapters used for iSCSI connectivity.
- When allocating LUNs or RAID groups for ESXi systems, multiple operating systems use and share that resource. As a result, the performance required from each LUN in the storage subsystem can be much higher if you are working with ESXi systems than if you are using physical machines. For example, if you expect to run four I/O intensive applications, allocate four times the performance capacity for the ESXi LUNs.
- When using multiple ESXi systems in conjunction with vCenter Server, the performance needed from the storage subsystem increases correspondingly.
- The number of outstanding I/Os needed by applications running on an ESXi system should match the number of I/Os the SAN can handle.

Network Performance

A typical SAN consists of a collection of computers connected to a collection of storage systems through a network of switches. Several computers often access the same storage.

Single Ethernet Link Connection to Storage shows several computer systems connected to a storage system through an Ethernet switch. In this configuration, each system is connected through a single Ethernet link to the switch, which is also connected to the storage system through a single Ethernet link. In most configurations, with modern switches and typical traffic, this is not a problem.

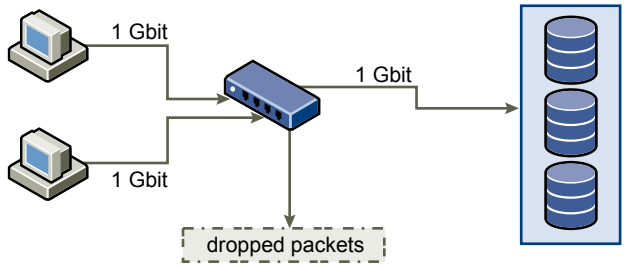
Figure 12-1. Single Ethernet Link Connection to Storage



When systems read data from storage, the maximum response from the storage is to send enough data to fill the link between the storage systems and the Ethernet switch. It is unlikely that any single system or virtual machine gets full use of the network speed, but this situation can be expected when many systems share one storage device.

When writing data to storage, multiple systems or virtual machines might attempt to fill their links. As Dropped Packets shows, when this happens, the switch between the systems and the storage system has to drop data. This happens because, while it has a single connection to the storage device, it has more traffic to send to the storage system than a single link can carry. In this case, the switch drops network packets because the amount of data it can transmit is limited by the speed of the link between it and the storage system.

Figure 12-2. Dropped Packets



Recovering from dropped network packets results in large performance degradation. In addition to time spent determining that data was dropped, the retransmission uses network bandwidth that could otherwise be used for current transactions.

iSCSI traffic is carried on the network by the Transmission Control Protocol (TCP). TCP is a reliable transmission protocol that ensures that dropped packets are retried and eventually reach their destination. TCP is designed to recover from dropped packets and retransmits them quickly and seamlessly. However, when the switch discards packets with any regularity, network throughput suffers significantly. The network becomes congested with requests to resend data and with the resent packets, and less data is actually transferred than in a network without congestion.

Most Ethernet switches can buffer, or store, data and give every device attempting to send data an equal chance to get to the destination. This ability to buffer some transmissions, combined with many systems limiting the number of outstanding commands, allows small bursts from several systems to be sent to a storage system in turn.

If the transactions are large and multiple servers are trying to send data through a single switch port, a switch's ability to buffer one request while another is transmitted can be exceeded. In this case, the switch drops the data it cannot send, and the storage system must request retransmission of the dropped packet. For example, if an Ethernet switch can buffer 32KB on an input port, but the server connected to it thinks it can send 256KB to the storage device, some of the data is dropped.

Most managed switches provide information on dropped packets, similar to the following:

```
*: interface is up
IHQ: pkts in input hold queue     IQD: pkts dropped from input queue
OHQ: pkts in output hold queue    OQD: pkts dropped from output queue
RXBS: rx rate (bits/sec)          RXPS: rx rate (pkts/sec)
TXBS: tx rate (bits/sec)          TXPS: tx rate (pkts/sec)
TRTL: throttle count
```

Table 12-1. Sample Switch Information

Interface	IHQ	IQD	OHQ	OQD	RXBS	RXPS	TXBS	TXPS	TRTL
* GigabitEthernet0/1	3	9922	0	0	476303000	62273	477840000	63677	0

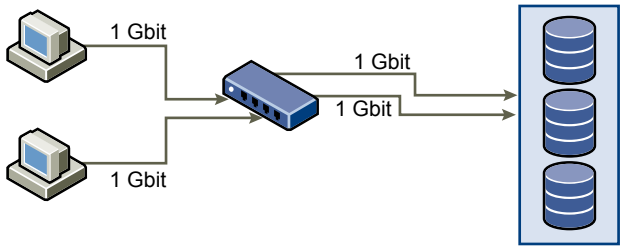
In this example from a Cisco switch, the bandwidth used is 476303000 bits/second, which is less than half of wire speed. In spite of this, the port is buffering incoming packets and has dropped quite a few packets. The final line of this interface summary indicates that this port has already dropped almost 10,000 inbound packets in the IQD column.

Configuration changes to avoid this problem involve making sure several input Ethernet links are not funneled into one output link, resulting in an oversubscribed link. When a number of links transmitting near capacity are switched to a smaller number of links, oversubscription is a possibility.

Generally, applications or systems that write a lot of data to storage, such as data acquisition or transaction logging systems, should not share Ethernet links to a storage device. These types of applications perform best with multiple connections to storage devices.

Multiple Connections from Switch to Storage shows multiple connections from the switch to the storage.

Figure 12-3. Multiple Connections from Switch to Storage



Using VLANs or VPNs does not provide a suitable solution to the problem of link oversubscription in shared configurations. VLANs and other virtual partitioning of a network provide a way of logically designing a network, but do not change the physical capabilities of links and trunks between switches. When storage traffic and other network traffic end up sharing physical connections, as they would with a VPN, the possibility for oversubscription and lost packets exists. The same is true of VLANs that share interswitch trunks. Performance design for a SANs must take into account the physical limitations of the network, not logical allocations.

Checking Ethernet Switch Statistics

Many Ethernet switches provide different methods for monitoring switch health.

Switches that have ports operating near maximum throughput much of the time do not provide optimum performance. If you have ports in your iSCSI SAN running near the maximum, reduce the load. If the port is connected to an ESXi system or iSCSI storage, you can reduce the load by using manual load balancing.

If the port is connected between multiple switches or routers, consider installing additional links between these components to handle more load. Ethernet switches also commonly provide information about transmission errors, queued packets, and dropped Ethernet packets. If the switch regularly reports any of these conditions on ports being used for iSCSI traffic, performance of the iSCSI SAN will be poor.

iSCSI SAN Configuration Checklist

This topic provides a checklist of special setup requirements for different storage systems and ESXi hosts.

Table 12-2. iSCSI SAN Configuration Requirements

Component	Comments
All storage systems	Write cache must be disabled if not battery backed.
Topology	No single failure should cause HBA and SP failover, especially with active-passive storage arrays.
EMC Symmetrix	Enable the SPC2 and SC3 settings. Contact EMC for the latest settings.
EMC Clariion	Set the EMC Clariion failover mode to 1 or 4. Contact EMC for details.
HP MSA	No specific requirements
HP EVA	For EVA3000/5000 firmware 4.001 and later, and EVA4000/6000/8000 firmware 5.031 and later, set the host type to `VMware`. Otherwise, set the host mode type to `Custom`. The value is: ■ EVA3000/5000 firmware 3.x: 000000002200282E ■ EVA4000/6000/8000: 000000202200083E
NetApp	If any of your iSCSI initiators are a part of an initiator group (igroup), disable ALUA on the NetApp array.

Table 12-2. iSCSI SAN Configuration Requirements (Continued)

Component	Comments
Dell EqualLogic	Make sure ARP Redirect is enabled on independent hardware iSCSI adapters.
HP StorageWorks (formerly LeftHand)	Make sure ARP Redirect is enabled on independent hardware iSCSI adapters.
ESXi Configuration	Set the following Advanced Settings for the ESXi host: ■ Set **Disk.UseLunReset** to **1** ■ Set **Disk.UseDeviceReset** to **0** A multipathing policy of Most Recently Used must be set for all LUNs hosting clustered disks for active-passive arrays. A multipathing policy of Most Recently Used or Fixed may be set for LUNs on active-active arrays. Allow ARP redirection if the storage system supports transparent failover.

13 Working with Datastores

Datastores are logical containers, analogous to file systems, that hide specifics of each storage device and provide a uniform model for storing virtual machine files. Datastores can also be used for storing ISO images, virtual machine templates, and floppy images.

You use the vSphere Client to access different types of storage devices that your ESXi host discovers and to deploy datastores on them.

Depending on the type of storage you use, datastores can be backed by the following file system formats:

- Virtual Machine File System (VMFS)
- Network File System (NFS)

After creating datastores, you can organize them in different ways. For example, you can group them into folders according to business practices. This allows you to assign the same permissions and alarms on the datastores in the group at one time.

You can also add datastores to datastore clusters. A datastore cluster is a collection of datastores with shared resources and a shared management interface. When you create a datastore cluster, you can use Storage DRS to manage storage resources. For information about datastore clusters, see the *vSphere Resource Management* documentation.

This chapter includes the following topics:

- "Understanding VMFS Datastores," on page 114
- "NFS Datastores," on page 127
- "Unmount VMFS or NFS Datastores," on page 128
- "Rename VMFS or NFS Datastores," on page 129
- "Group VMFS or NFS Datastores," on page 129
- "Handling Storage Device Disconnections," on page 130
- "Creating a Diagnostic Partition," on page 133
- "Set Up Dynamic Disk Mirroring," on page 134

Understanding VMFS Datastores

To store virtual disks, ESXi uses datastores, which are logical containers that hide specifics of physical storage from virtual machines and provide a uniform model for storing virtual machine files. Datastores that you deploy on block storage devices use the vSphere VMFS format, a special high-performance file system format that is optimized for storing virtual machines.

Several versions of the VMFS file system have been released since its introduction. The following table shows host-to-VMFS version relationships.

Table 13-1. Host access to VMFS version

VMFS	ESX/ESXi 3.x host	ESX/ESXi 4.x host	ESXi 5.0 host
VMFS2	RO	RO	N
VMFS3	RW	RW	RW
VMFS5	N	N	RW

- RW: Complete read and write support. You can create and power on virtual machines.
- RO: Read only support. You cannot create or power on virtual machines.
- N: No access. ESXi 5.0 does not support VMFS2. If your datastore was formatted with VMFS2, upgrade to VMFS5 using a two step process. For information, see "Upgrade VMFS2 Datastores to VMFS3," on page 121.

Use the vSphere Client to set up a VMFS datastore in advance on a block-based storage device that your ESXi host discovers. A VMFS datastore can be extended to span several physical storage extents, including SAN LUNs and local storage. This feature allows you to pool storage and gives you flexibility in creating the datastore necessary for your virtual machines.

You can increase the capacity of a datastore while virtual machines are running on the datastore. This ability lets you add new space to your VMFS datastores as your virtual machine requires it. VMFS is designed for concurrent access from multiple physical machines and enforces the appropriate access controls on virtual machine files.

How VMFS5 Differs from VMFS3

VMFS5 provides many improvements in scalability and performance over the previous version.

VMFS5 has the following improvements:

- Support of greater than 2TB storage devices for each VMFS extent.
- Increased resource limits such as file descriptors.
- Standard 1MB file system block size with support of 2TB virtual disks.
- Support of greater than 2TB disk size for RDMs in physical compatibility mode.
- Scalability improvements on storage devices that support hardware acceleration. For information, see Chapter 18, "Storage Hardware Acceleration," on page 173.
- Default use of hardware assisted locking, also called atomic test and set (ATS) locking, on storage devices that support hardware acceleration. For information about how to turn off ATS locking, see "Turn off ATS Locking," on page 127.
- Ability to reclaim physical storage space on thin provisioned storage devices. For information, see "Array Thin Provisioning and VMFS Datastores," on page 186.
- Online in-place upgrade process that upgrades existing datastores without disrupting hosts or virtual machines that are currently running.

VMFS Datastores and Storage Disk Formats

Storage devices that your host supports can use either the master boot record (MBR) format or the GUID partition table (GPT) format.

With ESXi 5.0, if you create a new VMFS5 datastore, the device is formatted with GPT. The GPT format enables you to create datastores larger than 2TB and up to 64TB.

VMFS3 datastores continue to use the MBR format for their storage devices. Consider the following items when you work with VMFS3 datastores:

- For VMFS3 datastores, the 2TB limit still applies, even when the storage device has a capacity of more than 2TB. To be able to use the entire storage space, upgrade a VMFS3 datastore to VMFS5. Conversion of the MBR format to GPT happens only after you expand the datastore.
- When you upgrade a VMFS3 datastore to VMFS5, any spanned extents have the GPT format.
- When you upgrade a VMFS3 datastore, remove from the storage device any partitions that ESXi does not recognize, for example, partitions that use the EXT2 or EXT3 formats. Otherwise, the host cannot format the device with GPT and the upgrade fails.
- You cannot expand a VMFS3 datastore on devices that have the GPT partition format.

VMFS Datastores as Repositories

ESXi can format SCSI-based storage devices as VMFS datastores. VMFS datastores primarily serve as repositories for virtual machines.

You can store multiple virtual machines on the same VMFS volume. Each virtual machine, encapsulated in a set of files, occupies a separate single directory. For the operating system inside the virtual machine, VMFS preserves the internal file system semantics, which ensures correct application behavior and data integrity for applications running in virtual machines.

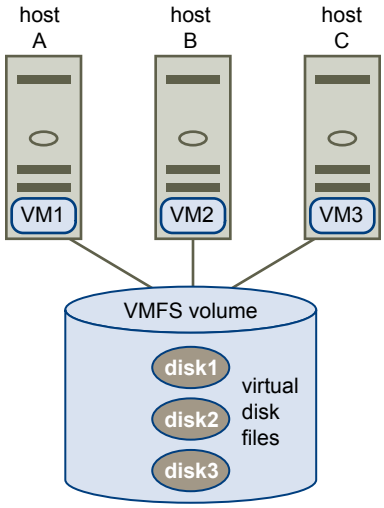
When you run multiple virtual machines, VMFS provides specific locking mechanisms for virtual machine files, so that virtual machines can operate safely in a SAN environment where multiple ESXi hosts share the same VMFS datastore.

In addition to virtual machines, the VMFS datastores can store other files, such as virtual machine templates and ISO images.

Sharing a VMFS Datastore Across Hosts

As a cluster file system, VMFS lets multiple ESXi hosts access the same VMFS datastore concurrently. You can connect up to 128 hosts to a single VMFS datastore.

Figure 13-1. Sharing a VMFS Datastore Across Hosts



To ensure that the same virtual machine is not accessed by multiple servers at the same time, VMFS provides on-disk locking.

Sharing the same VMFS volume across multiple hosts offers the following advantages:

- You can use VMware Distributed Resource Scheduling (DRS) and VMware High Availability (HA).

  You can distribute virtual machines across different physical servers. That means you run a mix of virtual machines on each server so that not all experience high demand in the same area at the same time. If a server fails, you can restart virtual machines on another physical server. In case of a failure, the on-disk lock for each virtual machine is released. For more information about VMware DRS, see the *vSphere Resource Management* documentation. For information about VMware HA, see the *vSphere Availability* documentation.

- You can use vMotion to migrate running virtual machines from one physical server to another. To support vMotion between ESXi 5 and version 3.x or 4.x hosts, virtual machines must be located on VMFS3 volumes. For information about migrating virtual machines, see the *vCenter Server and Host Management* documentation.

VMFS Metadata Updates

A VMFS datastore holds virtual machine files, directories, symbolic links, RDM descriptor files, and so on. The datastore also maintains a consistent view of all the mapping information for these objects. This mapping information is called metadata.

Metadata is updated each time you perform datastore or virtual machine management operations. Examples of operations requiring metadata updates include the following:

- Creating, growing, or locking a virtual machine file
- Changing a file's attributes
- Powering a virtual machine on or off
- Creating or deleting a VMFS datastore
- Expanding a VMFS datastore

- Creating a template
- Deploying a virtual machine from a template
- Migrating a virtual machine with vMotion

When metadata changes are made in a shared storage enviroment, VMFS uses special locking mechanisms to protect its data and prevent multiple hosts from concurrently writing to the metadata.

VMFS Locking Mechanisms

In a shared storage environment, when multiple hosts access the same VMFS datastore, specific locking mechanisms are used. These locking mechanism prevent multiple hosts from concurrently writing to the metadata and ensure that no data corruption occurs.

VMFS supports SCSI reservations and atomic test and set (ATS) locking.

SCSI Reservations

VMFS uses SCSI reservations on storage devices that do not support hardware acceleration. SCSI reservations lock an entire storage device while an operation that requires metadata protection is performed. After the operation completes, VMFS releases the reservation and other operations can continue. Because this lock is exclusive, excessive SCSI reservations by a host can cause performance degradation on other hosts that are accessing the same VMFS. For information about how to reduce SCSI reservations, see the *vSphere Troubleshooting* documentation.

Atomic Test and Set (ATS)

For storage devices that support hardware acceleration, VMFS uses the ATS algorithm, also called hardware assisted locking. In contrast with SCSI reservations, ATS supports discrete locking per disk sector. For information about hardware acceleration, see Chapter 18, "Storage Hardware Acceleration," on page 173.

Mechanisms that VMFS uses to apply different types of locking depend on the VMFS version.

Table 13-2. Use of ATS Locking on Devices with Hardware Acceleration Support

Storage Devices	New VMFS5	Upgraded VMFS5	VMFS3
Single extent	ATS only	ATS, but can revert to SCSI reservations	ATS, but can revert to SCSI reservations
Multiple extents	Spans only over ATS-capable devices	ATS except when locks on non-head	ATS except when locks on non-head

In certain cases, you might need to turn off the ATS-only setting for a new VMFS5 datastore. For information, see "Turn off ATS Locking," on page 127.

Creating and Increasing VMFS Datastores

You can set up VMFS datastores on any SCSI-based storage devices that your ESXi host discovers.

With VMFS5, you can have up to 256 VMFS datastores per host, with the maximum size of 64TB. The required minimum size for a VMFS datastore is 1.3GB, however, the recommended minimum size is 2GB.

NOTE Always have only one VMFS datastore for each LUN.

If your VMFS datastore requires more space, you can increase the VMFS volume. You can dynamically add new extents to any VMFS datastore. An extent is a partition on a physical storage device. The datastore can span over up to 32 extents with the size of each extent of more than 2TB, yet appear as a single volume.

NOTE ATS-only datastores cannot span over non-ATS devices.

Another option is to grow the existing datastore if the storage device where your datastore resides has free space.

Create a VMFS Datastore

VMFS datastores serve as repositories for virtual machines. You can set up VMFS datastores on any SCSI-based storage devices that the host discovers, including Fibre Channel, iSCSI, and local storage devices.

Prerequisites

Before creating datastores, you must install and configure any adapters that your storage requires. Rescan the adapters to discover newly added storage devices.

Procedure

1 Log in to the vSphere Client and select the host from the Inventory panel.

2 Click the **Configuration** tab and click **Storage** in the Hardware panel.

3 Click **Datastores** and click **Add Storage**.

4 Select the **Disk/LUN** storage type and click **Next**.

5 Select a device to use for your datastore and click **Next**.

IMPORTANT Select the device that does not have a datastore name displayed in the VMFS Label column. If a name is present, the device contains a copy of an existing VMFS datastore.

6 Select the **File System Version** and click **Next**.

IMPORTANT If you select VMFS3 you must select the maximum file size under **Formatting**.

7 If the disk is not blank, review the current disk layout in the top panel of the Current Disk Layout page and select a configuration option from the bottom panel.

Option	Description
Use all available partitions	Dedicates the entire disk to a single VMFS datastore. If you select this option, all file systems and data currently stored on this device are destroyed.
Use free space	Deploys a VMFS datastore in the remaining free space of the disk.

If the disk you are formatting is blank, the **Current Disk Layout** page presents the entire disk space for storage configuration.

8 Click **Next**.

9 On the **Properties** page, type a datastore name and click **Next**.

10 If the space specified for storage is excessive for your purposes, you can adjust the capacity values.

By default, the entire free space on the storage device is available.

11 Click **Next**.

12 In the Ready to Complete page, review the datastore configuration information and click **Finish**.

A datastore on the SCSI-based storage device is created. If you use the vCenter Server system to manage your hosts, the newly created datastore is added to all hosts.

Changing VMFS Datastore Properties

After you create a VMFS-based datastore, you can change its properties by using the Datastore Properties dialog box.

Depending on whether your vSphere Client is connected to a vCenter Server system or directly to a host, different ways to access the Datastore Properties dialog box exist.

Table 13-3. Access to the Datastore Properties Dialog Box

Connection Point	Actions
vCenter Server	1 Select **View > Inventory > Datastores and Datastore Clusters**. 2 Select the datastore form the inventory and click **Configuration > Properties**.
vCenter Server or ESXi host	1 Select a host from the inventory, click the **Configuration** tab and click **Storage**. 2 From the Datastores view, select the datastore to modify and click **Properties**.

Increase a VMFS Datastore

When you need to create virtual machines on a datastore, or when the virtual machines running on a datastore require more space, you can dynamically increase the capacity of a VMFS datastore.

Use one of the following methods to increase a VMFS datastore:

- Add a new extent. An extent is a partition on a storage device. You can add up to 32 extents of the same storage type to an existing VMFS datastore. The spanned VMFS datastore can use any or all of its extents at any time. It does not need to fill up a particular extent before using the next one.
- Grow an extent in an existing VMFS datastore, so that it fills the available adjacent capacity. Only extents with free space immediately after them are expandable.

NOTE If a shared datastore has powered on virtual machines and becomes 100% full, you can increase the datastore's capacity only from the host with which the powered on virtual machines are registered.

Prerequisites

Required privilege: **Host.Configuration.Storage Partition Configuration**

Procedure

1 Log in to the vSphere Client and select a host from the Inventory panel.

2 Click the **Configuration** tab and click **Storage**.

3 From the Datastores view, select the datastore to increase and click **Properties**.

4 Click **Increase**.

5 Select a device from the list of storage devices and click **Next**.

Option	Description
To add a new extent	Select the device for which the Expandable column reads NO.
To expand an existing extent	Select the device for which the Expandable column reads YES

6 Review the **Current Disk Layout** to see the available configurations and click **Next**.

7 Select a configuration option from the bottom panel.

Depending on the current layout of the disk and on your previous selections, the options you see might vary.

Option	Description
Use free space to add new extent	Adds the free space on this disk as a new extent.
Use free space to expand existing extent	Expands an existing extent to a required capacity.
Use free space	Deploys an extent in the remaining free space of the disk. This option is available only when you are adding an extent.
Use all available partitions	Dedicates the entire disk to a single extent. This option is available only when you are adding an extent and when the disk you are formatting is not blank. The disk is reformatted, and the datastores and any data that it contains are erased.

8 Set the capacity for the extent.

The minimum extent size is 1.3GB. By default, the entire free space on the storage device is available.

9 Click **Next**.

10 Review the proposed layout and the new configuration of your datastore, and click **Finish**.

What to do next

After you grow an extent in a shared VMFS datastore, refresh the datastore on each host that can access this datastore, so that the vSphere Client can display the correct datastore capacity for all hosts.

Upgrading VMFS Datastores

If your datastores were formatted with VMFS2 or VMFS3, you can upgrade the datastores to VMFS5.

When you perform datastore upgrades, consider the following items:

- To upgrade a VMFS2 datastore, you use a two-step process that involves upgrading VMFS2 to VMFS3 first. Because ESXi 5.0 hosts cannot access VMFS2 datastores, use a legacy host, ESX/ESXi 4.x or earlier, to access the VMFS2 datastore and perform the VMFS2 to VMFS3 upgrade.

  After you upgrade your VMFS2 datastore to VMFS3, the datastore becomes available on the ESXi 5.0 host, where you complete the process of upgrading to VMFS5.

- When you upgrade your datastore, the ESXi file-locking mechanism ensures that no remote host or local process is accessing the VMFS datastore being upgraded. Your host preserves all files on the datastore.

- The datastore upgrade is a one-way process. After upgrading your datastore, you cannot revert it back to its previous VMFS format.

An upgraded VMFS5 datastore differs from a newly formatted VMFS5.

Table 13-4. Comparing Upgraded and Newly Formatted VMFS5 Datastores

Characteristics	Upgraded VMFS5	Formatted VMFS5
File block size	1, 2, 4, and 8MB	1MB
Subblock size	64KB	8KB
Partition format	MBR. Conversion to GPT happens only after you expand the datastore.	GPT
Datastore limits	Retains limits of VMFS3 datastore.	

Upgrade VMFS2 Datastores to VMFS3

If your datastore was formatted with VMFS2, you must first upgrade it to VMFS3. Because ESXi 5.0 hosts cannot access VMFS2 datastores, use a legacy host, ESX/ESXi 4.x or earlier, to access the VMFS2 datastore and perform the VMFS2 to VMFS3 upgrade.

Prerequisites

- Commit or discard any changes to virtual disks in the VMFS2 datastore that you plan to upgrade.
- Back up the VMFS2 datastore.
- Be sure that no powered on virtual machines are using the VMFS2 datastore.
- Be sure that no other ESXi host is accessing the VMFS2 datastore.
- To upgrade the VMFS2 file system, its file block size must not exceed 8MB.

Procedure

1 Log in to the vSphere Client and select a host from the Inventory panel.

2 Click the **Configuration** tab and click **Storage**.

3 Select the datastore that uses the VMFS2 format.

4 Click **Upgrade to VMFS3**.

5 Perform a rescan on all hosts that see the datastore.

What to do next

After you upgrade your VMFS2 datastore to VMFS3, the datastore becomes available on the ESXi 5.0 host. You can now use the ESXi 5.0 host to complete the process of upgrading to VMFS5.

Upgrade VMFS3 Datastores to VMFS5

VMFS5 is a new version of the VMware cluster file system that provides performance and scalability improvements.

Prerequisites

- If you use a VMFS2 datastore, you must first upgrade it to VMFS3. Follow the instructions in "Upgrade VMFS2 Datastores to VMFS3," on page 121.
- All hosts accessing the datastore must support VMFS5.
- Verify that the volume to be upgraded has at least 2MB of free blocks available and 1 free file descriptor.

Procedure

1 Log in to the vSphere Client and select a host from the Inventory panel.

2 Click the **Configuration** tab and click **Storage**.

3 Select the VMFS3 datastore.

4 Click **Upgrade to VMFS5**.

A warning message about host version support appears.

5 Click **OK** to start the upgrade.

The task `Upgrade VMFS` appears in the **Recent Tasks** list.

6 Perform a rescan on all hosts that are associated with the datastore.

Managing Duplicate VMFS Datastores

When a storage device contains a VMFS datastore copy, you can mount the datastore with the existing signature or assign a new signature.

Each VMFS datastore created in a storage disk has a unique UUID that is stored in the file system superblock. When the storage disk is replicated or snapshotted, the resulting disk copy is identical, byte-for-byte, with the original disk. As a result, if the original storage disk contains a VMFS datastore with UUID X, the disk copy appears to contain an identical VMFS datastore, or a VMFS datastore copy, with exactly the same UUID X.

ESXi can detect the VMFS datastore copy and display it in the vSphere Client. You can mount the datastore copy with its original UUID or change the UUID, thus resignaturing the datastore.

In addition to LUN snapshotting and replication, the following storage device operations might cause ESXi to mark the existing datastore on the device as a copy of the original datastore:

- LUN ID changes
- SCSI device type changes, for example, from SCSI-2 to SCSI-3
- SPC-2 compliancy enablement

Mount a VMFS Datastore with an Existing Signature

If you do not need to resignature a VMFS datastore copy, you can mount it without changing its signature.

You can keep the signature if, for example, you maintain synchronized copies of virtual machines at a secondary site as part of a disaster recovery plan. In the event of a disaster at the primary site, you mount the datastore copy and power on the virtual machines at the secondary site.

Important You can mount a VMFS datastore copy only if it does not collide with the original VMFS datastore that has the same UUID. To mount the copy, the original VMFS datastore has to be offline.

When you mount the VMFS datastore, ESXi allows both reads and writes to the datastore residing on the LUN copy. The LUN copy must be writable. The datastore mounts are persistent and valid across system reboots.

Prerequisites

Before you mount a VMFS datastore, perform a storage rescan on your host so that it updates its view of LUNs presented to it.

Procedure

1. Log in to the vSphere Client and select the server from the inventory panel.
2. Click the **Configuration** tab and click **Storage** in the Hardware panel.
3. Click **Add Storage**.
4. Select the **Disk/LUN** storage type and click **Next**.
5. From the list of LUNs, select the LUN that has a datastore name displayed in the VMFS Label column and click **Next**.

   The name present in the VMFS Label column indicates that the LUN is a copy that contains a copy of an existing VMFS datastore.
6. Under Mount Options, select **Keep Existing Signature**.
7. In the Ready to Complete page, review the datastore configuration information and click **Finish**.

What to do next

If you later want to resignature the mounted datastore, you must unmount it first.

Resignature a VMFS Datastore Copy

Use datastore resignaturing if you want to retain the data stored on the VMFS datastore copy.

When resignaturing a VMFS copy, ESXi assigns a new UUID and a new label to the copy, and mounts the copy as a datastore distinct from the original.

The default format of the new label assigned to the datastore is `snap-`*`snapID`*`-`*`oldLabel`*, where *snapID* is an integer and *oldLabel* is the label of the original datastore.

When you perform datastore resignaturing, consider the following points:

- Datastore resignaturing is irreversible.
- The LUN copy that contains the VMFS datastore that you resignature is no longer treated as a LUN copy.
- A spanned datastore can be resignatured only if all its extents are online.
- The resignaturing process is crash and fault tolerant. If the process is interrupted, you can resume it later.
- You can mount the new VMFS datastore without a risk of its UUID colliding with UUIDs of any other datastore, such as an ancestor or child in a hierarchy of LUN snapshots.

Prerequisites

To resignature a mounted datastore copy, first unmount it.

Before you resignature a VMFS datastore, perform a storage rescan on your host so that the host updates its view of LUNs presented to it and discovers any LUN copies.

Procedure

1 Log in to the vSphere Client and select the server from the inventory panel.

2 Click the **Configuration** tab and click **Storage** in the Hardware panel.

3 Click **Add Storage**.

4 Select the **Disk/LUN** storage type and click **Next**.

5 From the list of LUNs, select the LUN that has a datastore name displayed in the VMFS Label column and click **Next**.

The name present in the VMFS Label column indicates that the LUN is a copy that contains a copy of an existing VMFS datastore.

6 Under Mount Options, select **Assign a New Signature** and click **Next**.

7 In the Ready to Complete page, review the datastore configuration information and click **Finish**.

What to do next

After resignaturing, you might have to do the following:

- If the resignatured datastore contains virtual machines, update references to the original VMFS datastore in the virtual machine files, including `.vmx`, `.vmdk`, `.vmsd`, and `.vmsn`.
- To power on virtual machines, register them with vCenter Server.

Delete VMFS Datastores

You can delete any type of VMFS datastore, including copies that you have mounted without resignaturing. When you delete a datastore, it is destroyed and disappears from all hosts that have access to the datastore.

NOTE The datastore delete operation permanently deletes all files associated with virtual machines on the datastore.

Prerequisites

- Remove all virtual machines from the datastore.
- Make sure that no other host is accessing the datastore.

Procedure

1 Display the datastores.

2 Right-click the datastore to delete and click **Delete**.

3 Confirm that you want to delete the datastore.

Storage Refresh and Rescan Operations

The refresh operation for datastores, storage devices, and storage adapters updates the lists and storage information displayed in the vSphere Client. For example, it updates such information as the datastore capacity. When you perform storage management tasks or make changes in the SAN configuration, you might need to rescan your storage.

When you perform VMFS datastore management operations, such as creating a VMFS datastore or RDM, adding an extent, and increasing or deleting a VMFS datastore, your host or the vCenter Server automatically rescans and updates your storage. You can disable the automatic rescan feature by turning off the Host Rescan Filter. See "Turn off Storage Filters," on page 125.

In certain cases, you need to perform a manual rescan. You can rescan all storage available to your host, or, if you are using the vCenter Server, to all hosts in a folder, cluster, and datacenter.

If the changes you make are isolated to storage connected through a specific adapter, perform a rescan for this adapter.

Perform the manual rescan each time you make one of the following changes.

- Zone a new disk array on a SAN.
- Create new LUNs on a SAN.
- Change the path masking on a host.
- Reconnect a cable.
- Change CHAP settings (iSCSI only).
- Add or remove discovery or static addresses (iSCSI only).
- Add a single host to the vCenter Server after you have edited or removed from the vCenter Server a datastore shared by the vCenter Server hosts and the single host.

IMPORTANT If you rescan when a path is unavailable, the host removes the path from the list of paths to the device. The path reappears on the list as soon as it becomes available and starts working again.

Perform Storage Rescan

When you make changes in your SAN configuration, you might need to rescan your storage. You can rescan all storage available to your host. If the changes you make are isolated to storage accessed through a specific adapter, perform rescan for only this adapter.

Use this procedure if you want to limit the rescan to storage available to a particular host or accessed through a particular adapter on the host. If you want to rescan storage available to all hosts managed by your vCenter Server system, you can do so by right-clicking a datacenter, cluster, or folder that contains the hosts and selecting **Rescan for Datastores**.

Procedure

1 In the vSphere Client, select a host and click the **Configuration** tab.

2 Select a rescan option.

Option	Description
Storage	In the Hardware panel, click **Storage**, and click **Rescan All** above the Datastores or Devices panel.
Storage Adapters	In the Hardware panel, click **Storage Adapters**, and click **Rescan All** above the Storage Adapters panel. NOTE You can also right-click an individual adapter and select **Rescan** to rescan just that adapter.

3 Specify extent of rescan.

Option	Description
Scan for New Storage Devices	Rescan all adapters to discover new storage devices. If new devices are discovered, they appear in the device list.
Scan for New VMFS Volumes	Rescan all storage devices to discover new datastores that have been added since the last scan. Any new datastores appear in the datastore list.

Change the Number of Scanned LUNs

By default, the VMkernel scans for LUN 0 to LUN 255 for every target (a total of 256 LUNs). You can modify the **Disk.MaxLUN** parameter to improve LUN discovery speed.

IMPORTANT You cannot discover LUNs with a LUN ID number that is greater than 255.

Reducing the value can shorten rescan time and boot time. However, the time to rescan LUNs might depend on other factors, including the type of storage system and whether sparse LUN support is enabled.

Procedure

1 In the vSphere Client inventory panel, select the host, click the **Configuration** tab, and click **Advanced Settings** under Software.

2 Select **Disk**.

3 Scroll down to **Disk.MaxLUN**.

4 Change the existing value to the value of your choice, and click **OK**.

The value you enter specifies the LUN after the last one you want to discover.

For example, to discover LUNs from 0 through 31, set **Disk.MaxLUN** to 32.

Turn off Storage Filters

When you perform VMFS datastore management operations, vCenter Server uses default storage protection filters. The filters help you to avoid storage corruption by retrieving only the storage devices that can be used for a particular operation. Unsuitable devices are not displayed for selection. You can turn off the filters to view all devices.

Before making any changes to the device filters, consult with the VMware support team. You can turn off the filters only if you have other methods to prevent device corruption.

Procedure

1 In the vSphere Client, select **Administration > vCenter Server Settings**.

2 In the settings list, select **Advanced Settings**.

3 In the **Key** text box, type a key.

Key	Filter Name
config.vpxd.filter.vmfsFilter	VMFS Filter
config.vpxd.filter.rdmFilter	RDM Filter
config.vpxd.filter.SameHostAndTransportsFilter	Same Host and Transports Filter
config.vpxd.filter.hostRescanFilter	Host Rescan Filter **NOTE** If you turn off the Host Rescan Filter, your hosts continue to perform a rescan each time you present a new LUN to a host or a cluster.

4 In the **Value** text box, type **`False`** for the specified key.

5 Click **Add**.

6 Click **OK**.

You are not required to restart the vCenter Server system.

Storage Filtering

vCenter Server provides storage filters to help you avoid storage device corruption or performance degradation that can be caused by an unsupported use of storage devices. These filters are available by default.

Table 13-5. Storage Filters

Filter Name	Description	Key
VMFS Filter	Filters out storage devices, or LUNs, that are already used by a VMFS datastore on any host managed by vCenter Server. The LUNs do not show up as candidates to be formatted with another VMFS datastore or to be used as an RDM.	config.vpxd.filter.vmfsFilter
RDM Filter	Filters out LUNs that are already referenced by an RDM on any host managed by vCenter Server. The LUNs do not show up as candidates to be formatted with VMFS or to be used by a different RDM. If you need virtual machines to access the same LUN, the virtual machines must share the same RDM mapping file. For information about this type of configuration, see the *vSphere Resource Management* documentation.	config.vpxd.filter.rdmFilter
Same Host and Transports Filter	Filters out LUNs ineligible for use as VMFS datastore extents because of host or storage type incompatibility. Prevents you from adding the following LUNs as extents: ■ LUNs not exposed to all hosts that share the original VMFS datastore. ■ LUNs that use a storage type different from the one the original VMFS datastore uses. For example, you cannot add a Fibre Channel extent to a VMFS datastore on a local storage device.	config.vpxd.filter.SameHostAndTransportsFilter
Host Rescan Filter	Automatically rescans and updates VMFS datastores after you perform datastore management operations. The filter helps provide a consistent view of all VMFS datastores on all hosts managed by vCenter Server. **NOTE** If you present a new LUN to a host or a cluster, the hosts automatically perform a rescan no matter whether you have the Host Rescan Filter on or off.	config.vpxd.filter.hostRescanFilter

Turn off ATS Locking

When you create a VMFS5 datastore on a device that supports atomic test and set (ATS) locking, the datastore is set to the ATS-only mode. In certain circumstances, you might need to turn off the ATS mode setting.

Turn off the ATS setting when, for example, your storage device is downgraded or firmware updates fail and the device no longer supports hardware acceleration. The option that you use to turn off the ATS setting is available only through the ESXi Shell. For more information, see the *Getting Started with vSphere Command-Line Interfaces*.

Procedure

- To turn off the ATS setting, run the following command:

  ```
  vmkfstools –-configATSOnly 0 device
  ```

  The *device* parameter is the path to the head extent device on which VMFS5 was deployed. Use the following format:

  ```
  /vmfs/devices/disks/disk_ID:P
  ```

NFS Datastores

ESXi can access a designated NFS volume located on a NAS server, mount the volume, and use it for its storage needs. You can use NFS volumes to store and boot virtual machines in the same way that you use VMFS datastores.

ESXi supports the following shared storage capabilities on NFS volumes:

- vMotion
- VMware DRS and VMware HA
- ISO images, which are presented as CD-ROMs to virtual machines
- Virtual machine snapshots

When you work with NFS storage, the following considerations apply:

- The maximum size of NFS datastores depends on the support that an NFS server provides. ESXi does not impose any limits on the NFS datastore size.
- If you use non-ASCII characters to name datastores and virtual machines, make sure that the underlying NFS server offers internationalization support. If the server does not support international characters, use only ASCII characters, otherwise unpredictable failures might occur.

NFS Datastores as Repositories for Commonly Used Files

In addition to storing virtual disks on NFS datastores, you can also use NFS as a central repository for ISO images, virtual machine templates, and so on.

To use NFS as a shared repository, you create a directory on the NFS server and then mount it as a datastore on all hosts. If you use the datastore for ISO images, you can connect the virtual machine's CD-ROM device to an ISO file on the datastore and install a guest operating system from the ISO file.

NOTE If the underlying NFS volume, on which the files are stored, is read-only, make sure that the volume is exported as a read-only share by the NFS server, or configure it as a read-only datastore on the ESXi host. Otherwise, the host considers the datastore to be read-write and might not be able to open the files.

Create NFS Datastores

You can use the Add Storage wizard to mount an NFS volume and use it as if it were a VMFS datastore.

Prerequisites

Because NFS requires network connectivity to access data stored on remote servers, before configuring NFS, you must first configure VMkernel networking.

Procedure

1 Log in to the vSphere Client and select the host from the Inventory panel.

2 Click the **Configuration** tab and click **Storage** in the Hardware panel.

3 Click **Datastores** and click **Add Storage**.

4 Select **Network File System** as the storage type and click **Next**.

5 Enter the server name, the mount point folder name, and the datastore name.

For the server name, you can enter an IP address, a DNS name, or an NFS UUID.

NOTE When you mount the same NFS volume on different hosts, make sure that the server and folder names are identical across the hosts. If the names do not match exactly, the hosts see the same NFS volume as two different datastores. This might result in a failure of such features as vMotion. An example of such discrepancy could be if you enter `filer` as the server name on one host and `filer.domain.com` on the other.

6 (Optional) Select **Mount NFS read only** if the volume is exported as read only by the NFS server.

7 Click **Next**.

8 In the Network File System Summary page, review the configuration options and click **Finish**.

Unmount VMFS or NFS Datastores

When you unmount a datastore, it remains intact, but can no longer be seen from the hosts that you specify. The datastore continues to appear on other hosts, where it remains mounted.

Do not perform any configuration operations that might result in I/O to the datastore while the unmount is in progress.

NOTE vSphere HA heartbeating does not prevent you from unmounting the datastore. If a datastore is used for heartbeating, unmounting it might cause the host to fail and restart any active virtual machine. If the heartbeating check fails, the vSphere Client displays a warning.

Prerequisites

Before unmounting VMFS datastores, make sure that the following prerequisites are met:

- No virtual machines reside on the datastore.
- The datastore is not part of a datastore cluster.
- The datastore is not managed by Storage DRS.
- Storage I/O control is disabled for this datastore.
- The datastore is not used for vSphere HA heartbeating.

Procedure

1 Display the datastores.

2 Right-click the datastore to unmount and select **Unmount**.

3 If the datastore is shared, specify which hosts should no longer access the datastore.

   a Deselect the hosts on which you want to keep the datastore mounted.

   By default, all hosts are selected.

   b Click **Next**.

   c Review the list of hosts from which to unmount the datastore, and click **Finish**.

4 Confirm that you want to unmount the datastore.

After you unmount a VMFS datastore, the datastore becomes inactive and is dimmed in the host's datastore list. An unmounted NFS datastore no longer appears on the list.

NOTE The datastore that is unmounted from some hosts while being mounted on others, is shown as active in the Datastores and Datastore Clusters view.

Rename VMFS or NFS Datastores

You can change the name of an existing datastore.

Procedure

1 Display the datastores.

2 Right-click the datastore to rename and select **Rename**.

3 Type a new datastore name.

If you use the vCenter Server system to manage your hosts, the new name appears on all hosts that have access to the datastore.

Group VMFS or NFS Datastores

If you use the vCenter Server system to manage your hosts, group datastores into folders. This allows you to organize your datastores according to business practices and to assign the same permissions and alarms on the datastores in the group at one time.

Procedure

1 Log in to the vSphere Client.

2 If necessary, create the datastores.

3 In the Inventory panel, choose **Datastores**.

4 Select the datacenter containing the datastores to group.

5 In the shortcut menu, click the **New Folder** icon.

6 Give the folder a descriptive name.

7 Click and drag each datastore onto the folder.

Handling Storage Device Disconnections

ESXi supports planned device removal and the detection of unplanned device loss.

Your host can determine whether a device disconnection is a temporary, all-paths-down event or whether a permanent device disconnection occurred. Planned device removal is an intentional disconnection of a storage device that uses the ability to unmount VMFS datastores. You can perform a storage reconfiguration in which you detach a datastore and later reattach it. If your host detects an unplanned device loss, ESXi marks the storage device as permanently unavailable to conserve resources and memory.

- Planned Device Removal on page 130

  Planned device removal is a device disconnection detectable by an ESXi host. You can perform an orderly removal and reconnection of a storage device.

- Unplanned Device Loss on page 132

  Unplanned device loss is a condition that occurs when your ESXi host permanently loses connection to a storage device.

Planned Device Removal

Planned device removal is a device disconnection detectable by an ESXi host. You can perform an orderly removal and reconnection of a storage device.

Planned device removal is the intentional disconnection of a storage device. You might plan to remove a device for a variety of reasons, such as upgrading your hardware or reconfiguring your storage devices. To perform an orderly removal and reconnection of a storage device, use the following procedure:

1 Migrate virtual machines from the device you plan to detach.

  See the *vCenter Server and Host Management* documentation.

2 Unmount the datastore deployed on the device.

  See "Unmount VMFS or NFS Datastores," on page 128.

3 Detach the storage device.

  See "Detach Storage Devices," on page 130.

  You can now perform a reconfiguration of the storage device by using the array console.

4 Reattach the storage device.

  See "Attach Storage Devices," on page 131.

5 Mount the datastore and restart the virtual machines.

  - To mount a shared datastore, see "Mount Shared VMFS Datastores," on page 131.
  - To mount an unshared datastore, see "Mount Unshared VMFS Datastores," on page 131.

Detach Storage Devices

Use the vSphere Client to safely detach a storage device from your host.

You might need to detach the device to make it inaccessible to your host, when, for example, you perform a hardware upgrade on the storage side.

Prerequisites

- The device does not contain any datastores.
- No virtual machines use the device as an RDM disk.

- The device does not contain a diagnostic partition.

Procedure

1 In the vSphere Client, display storage devices.

2 Right-click the device to detach and select **Detach**.

The device name is dimmed in the vSphere Client and becomes inaccessible. The operational state of the device changes to Unmounted.

What to do next

If multiple hosts share the device, detach the device from each host.

Attach Storage Devices

Use the vSphere Client to reattach a storage device that you previously detached.

Procedure

1 In the vSphere Client, display storage devices.

2 Right-click the detached storage device and select **Attach**.

The device becomes accessible.

Mount Shared VMFS Datastores

You mount an unmounted VMFS datastore by using the context menu.

You can use the context menu to mount a datastore after you unmount a datastore from a host. If the datastore was accessible from more than one host before it was unmounted, you can use the Mount Datastore wizard to select the hosts that the datastore will be mounted on.

Procedure

1 Select **Home > Inventory > Datastores and Datastore Cluster**.

2 Right-click an unmounted datastore and select **Mount**.

NOTE The datastore that is unmounted from some hosts while being mounted on others, is shown as active in the Datastores and Datastore Clusters view.

3 Specify which hosts should access the datastore in the Mount Datastore wizard.

By default, all hosts are selected.

a Deselect the hosts where you want to keep the datastore unmounted.

b Click **Next**.

c Review the list of hosts on which to mount the datastore and click **Finish**.

Mount Unshared VMFS Datastores

Use this procedure to mount a previously unmounted datastore on a single host.

Procedure

1 Log in to the vSphere Client and select the host on which to mount the unmounted datastore.

NOTE If the datastore is shared, it will be mounted only on the host you select. It will remain unmounted on other hosts not specified in this procedure.

2 Click the **Configuration** tab and click **Storage** in the Hardware panel.

3 Right-click the datastore and select **Mount**.

NOTE The datastore that is mounted on some hosts while being unmounted from others, is shown as active in the Datastores and Datastore Clusters view of the vCenter Server.

Unplanned Device Loss

Unplanned device loss is a condition that occurs when your ESXi host permanently loses connection to a storage device.

Problem

Although the ESXi host cannot determine the reason for a device loss, the host supports the loss detection. When the device becomes permanently unavailable, ESXi receives sense codes from storage arrays and recognizes that the device is permanently lost, not just temporarily unavailable. ESXi marks the device as not connected and a warning about the device being permanently unavailable appears in the VMkernel log file.

Cause

Typically, unplanned device loss is unintentional and can occur when a storage device is unmapped, removed, or its unique ID changes, or when there is an unrecoverable hardware error.

Solution

If you experience an unplanned device loss condition, you must unmount any related datastores and perform a storage rescan to remove the persistent information associated with the device. See the following topics:

- To verify the status of the device, see "Check the Connection Status of a Storage Device," on page 132.
- To unmount a datastore, see "Unmount VMFS or NFS Datastores," on page 128.
- To perform a rescan, see "Perform Storage Rescan," on page 124.

Check the Connection Status of a Storage Device

Use the `esxcli` command to verify the connection status of a particular storage device.

In the procedure, **`--server=`*`server_name`*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

- Run the **`esxcli --server=`*`server_name`*` storage core device list -d=`*`device_ID`*** command.

  The following sample output shows that the device is not connected.

```
# esxcli --server=server_name storage core device list -d naa.XXXXXXXXXXXX4c
naa.XXXXXXXXXXX4c
 Display Name: XXXX Fibre Channel Disk(naa.XXXXXXXXXXX4c)
 Size: 20480
 Device Type: Direct-Access
 Multipath Plugin: NMP
 XXXXXXXXXXXXXXX
 Status: not connected
 XXXXXXXXXXXXXXX
```

Creating a Diagnostic Partition

To run successfully, your host must have a diagnostic partition or a dump partition to store core dumps for debugging and technical support.

Typically, a local diagnostic partition is created during ESXi installation. You can override this default behavior if, for example, you use shared storage devices instead of local storage. To prevent automatic disk formatting, detach the local storage devices from the host before you install ESXi and power on the host for the first time. You can later create a diagnostic partition on a local disk or on a private or shared SAN LUN using the vSphere Client.

The following considerations apply:

- A diagnostic partition cannot be located on an iSCSI LUN accessed through the software iSCSI or dependent hardware iSCSI adapter. For more information about diagnostic partitions with iSCSI, see "General Boot from iSCSI SAN Recommendations," on page 97.
- Unless you are using diskless servers, set up a diagnostic partition on a local storage.
- Each host must have a diagnostic partition of 110MB. If multiple hosts share a diagnostic partition on a SAN LUN, the partition should be large enough to accommodate core dumps of all hosts.
- If a host that uses a shared diagnostic partition fails, reboot the host and extract log files immediately after the failure. Otherwise, the second host that fails before you collect the diagnostic data of the first host might not be able to save the core dump.

To mange the host's diagnostic partition, use the vCLI commands. See *vSphere Command-Line Interface Concepts and Examples*.

Create a Diagnostic Partition

You can create a diagnostic partition for your host.

Procedure

1 Log in to the vSphere Client and select the host from the Inventory panel.

2 Click the **Configuration** tab and click **Storage** in the Hardware panel.

3 Click **Datastores** and click **Add Storage**.

4 Select **Diagnostic** and click **Next**.

If you do not see **Diagnostic** as an option, the host already has a diagnostic partition.

5 Specify the type of diagnostic partition.

Option	Description
Private Local	Creates the diagnostic partition on a local disk. This partition stores fault information only for your host.
Private SAN Storage	Creates the diagnostic partition on a non-shared SAN LUN. This partition stores fault information only for your host.
Shared SAN Storage	Creates the diagnostic partition on a shared SAN LUN. This partition is accessed by multiple hosts and can store fault information for more than one host.

6 Click **Next**.

7 Select the device to use for the diagnostic partition and click **Next**.

8 Review the partition configuration information and click **Finish**.

Set Up Dynamic Disk Mirroring

Typically, you cannot use logical-volume manager software on virtual machines to mirror virtual disks. However, if your Microsoft Windows virtual machines support dynamic disks, you can protect the virtual machines from an unplanned storage device loss by mirroring virtual disks across two SAN LUNs.

Prerequisites

- Use a Windows virtual machine that supports dynamic disks.
- Required privilege: **Advanced**

Procedure

1 Create a virtual machine with two virtual disks.

  Make sure to place the disks on different datastores.

2 Log in to your virtual machine and configure the disks as dynamic mirrored disks.

  See Microsoft documentation.

3 After the disks synchronise, power off the virtual machine.

4 Change virtual machine settings to allow the use of dynamic disk mirroring.

  a Right-click the virtual machine and select **Edit Settings**.

  b Click the **Options** tab and under **Advanced**, select **General**.

  c Click **Configuration Parameters**.

  d Click **Add Row** and add the following parameters:

Name	Value
scsi#.returnNoConnectDuringAPD	True
scsi#.returnBusyOnNoConnectStatus	False

  e Click **OK**.

14 Raw Device Mapping

Raw device mapping (RDM) provides a mechanism for a virtual machine to have direct access to a LUN on the physical storage subsystem (Fibre Channel or iSCSI only).

The following topics contain information about RDMs and provide instructions on how to create and manage RDMs.

This chapter includes the following topics:

- "About Raw Device Mapping," on page 135
- "Raw Device Mapping Characteristics," on page 138
- "Create Virtual Machines with RDMs," on page 140
- "Manage Paths for a Mapped Raw LUN," on page 141

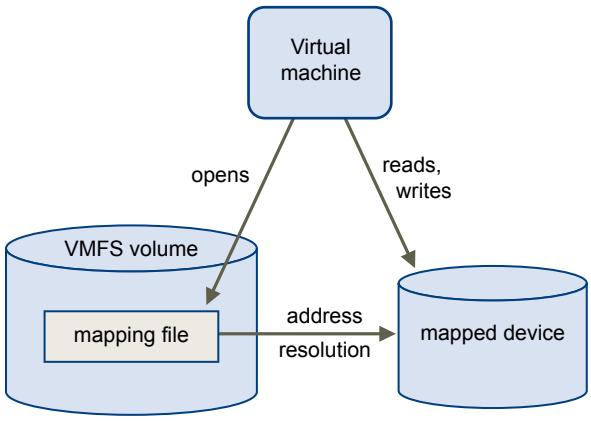
About Raw Device Mapping

An RDM is a mapping file in a separate VMFS volume that acts as a proxy for a raw physical storage device. The RDM allows a virtual machine to directly access and use the storage device. The RDM contains metadata for managing and redirecting disk access to the physical device.

The file gives you some of the advantages of direct access to a physical device while keeping some advantages of a virtual disk in VMFS. As a result, it merges VMFS manageability with raw device access.

RDMs can be described in terms such as mapping a raw device into a datastore, mapping a system LUN, or mapping a disk file to a physical disk volume. All these terms refer to RDMs.

Figure 14-1. Raw Device Mapping



Although VMware recommends that you use VMFS datastores for most virtual disk storage, on certain occasions, you might need to use raw LUNs or logical disks located in a SAN.

For example, you need to use raw LUNs with RDMs in the following situations:

- When SAN snapshot or other layered applications run in the virtual machine. The RDM better enables scalable backup offloading systems by using features inherent to the SAN.
- In any MSCS clustering scenario that spans physical hosts — virtual-to-virtual clusters as well as physical-to-virtual clusters. In this case, cluster data and quorum disks should be configured as RDMs rather than as virtual disks on a shared VMFS.

Think of an RDM as a symbolic link from a VMFS volume to a raw LUN. The mapping makes LUNs appear as files in a VMFS volume. The RDM, not the raw LUN, is referenced in the virtual machine configuration. The RDM contains a reference to the raw LUN.

Using RDMs, you can:

- Use vMotion to migrate virtual machines using raw LUNs.
- Add raw LUNs to virtual machines using the vSphere Client.
- Use file system features such as distributed file locking, permissions, and naming.

Two compatibility modes are available for RDMs:

- Virtual compatibility mode allows an RDM to act exactly like a virtual disk file, including the use of snapshots.
- Physical compatibility mode allows direct access of the SCSI device for those applications that need lower level control.

Benefits of Raw Device Mapping

An RDM provides a number of benefits, but it should not be used in every situation. In general, virtual disk files are preferable to RDMs for manageability. However, when you need raw devices, you must use the RDM.

RDM offers several benefits.

User-Friendly Persistent Names
Provides a user-friendly name for a mapped device. When you use an RDM, you do not need to refer to the device by its device name. You refer to it by the name of the mapping file, for example:

```
/vmfs/volumes/myVolume/myVMDirectory/myRawDisk.vmdk
```

Dynamic Name Resolution
Stores unique identification information for each mapped device. VMFS associates each RDM with its current SCSI device, regardless of changes in the physical configuration of the server because of adapter hardware changes, path changes, device relocation, and so on.

Distributed File Locking
Makes it possible to use VMFS distributed locking for raw SCSI devices. Distributed locking on an RDM makes it safe to use a shared raw LUN without losing data when two virtual machines on different servers try to access the same LUN.

File Permissions
Makes file permissions possible. The permissions of the mapping file are enforced at file-open time to protect the mapped volume.

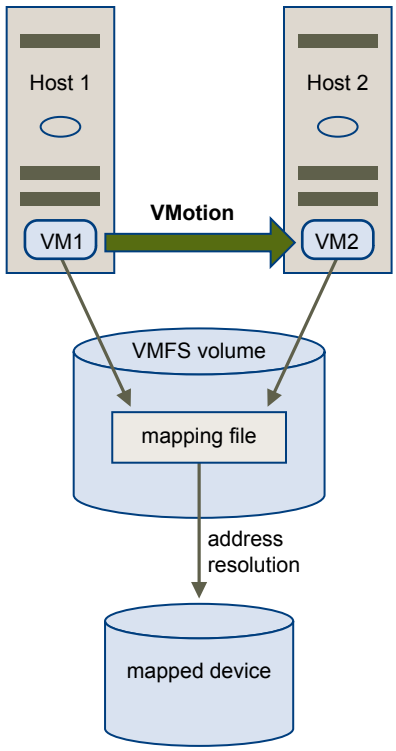
File System Operations
Makes it possible to use file system utilities to work with a mapped volume, using the mapping file as a proxy. Most operations that are valid for an ordinary file can be applied to the mapping file and are redirected to operate on the mapped device.

Snapshots
Makes it possible to use virtual machine snapshots on a mapped volume. Snapshots are not available when the RDM is used in physical compatibility mode.

vMotion

Lets you migrate a virtual machine with vMotion. The mapping file acts as a proxy to allow vCenter Server to migrate the virtual machine by using the same mechanism that exists for migrating virtual disk files.

Figure 14-2. vMotion of a Virtual Machine Using Raw Device Mapping



SAN Management Agents

Makes it possible to run some SAN management agents inside a virtual machine. Similarly, any software that needs to access a device by using hardware-specific SCSI commands can be run in a virtual machine. This kind of software is called SCSI target-based software. When you use SAN management agents, select a physical compatibility mode for the RDM.

N-Port ID Virtualization (NPIV)

Makes it possible to use the NPIV technology that allows a single Fibre Channel HBA port to register with the Fibre Channel fabric using several worldwide port names (WWPNs). This ability makes the HBA port appear as multiple virtual ports, each having its own ID and virtual port name. Virtual machines can then claim each of these virtual ports and use them for all RDM traffic.

NOTE You can use NPIV only for virtual machines with RDM disks.

VMware works with vendors of storage management software to ensure that their software functions correctly in environments that include ESXi. Some applications of this kind are:

- SAN management software
- Storage resource management (SRM) software
- Snapshot software
- Replication software

Such software uses a physical compatibility mode for RDMs so that the software can access SCSI devices directly.

Various management products are best run centrally (not on the ESXi machine), while others run well on the virtual machines. VMware does not certify these applications or provide a compatibility matrix. To find out whether a SAN management application is supported in an ESXi environment, contact the SAN management software provider.

Limitations of Raw Device Mapping

Certain limitations exist when you use RDMs.

- The RDM is not available for direct-attached block devices or certain RAID devices. The RDM uses a SCSI serial number to identify the mapped device. Because block devices and some direct-attach RAID devices do not export serial numbers, they cannot be used with RDMs.
- If you are using the RDM in physical compatibility mode, you cannot use a snapshot with the disk. Physical compatibility mode allows the virtual machine to manage its own, storage-based, snapshot or mirroring operations.

  Virtual machine snapshots are available for RDMs with virtual compatibility mode.
- You cannot map to a disk partition. RDMs require the mapped device to be a whole LUN.

Raw Device Mapping Characteristics

An RDM is a special mapping file in a VMFS volume that manages metadata for its mapped device. The mapping file is presented to the management software as an ordinary disk file, available for the usual file-system operations. To the virtual machine, the storage virtualization layer presents the mapped device as a virtual SCSI device.

Key contents of the metadata in the mapping file include the location of the mapped device (name resolution), the locking state of the mapped device, permissions, and so on.

RDM Virtual and Physical Compatibility Modes

You can use RDMs in virtual compatibility or physical compatibility modes. Virtual mode specifies full virtualization of the mapped device. Physical mode specifies minimal SCSI virtualization of the mapped device, allowing the greatest flexibility for SAN management software.

In virtual mode, the VMkernel sends only READ and WRITE to the mapped device. The mapped device appears to the guest operating system exactly the same as a virtual disk file in a VMFS volume. The real hardware characteristics are hidden. If you are using a raw disk in virtual mode, you can realize the benefits of VMFS such as advanced file locking for data protection and snapshots for streamlining development processes. Virtual mode is also more portable across storage hardware than physical mode, presenting the same behavior as a virtual disk file.

In physical mode, the VMkernel passes all SCSI commands to the device, with one exception: the REPORT LUNs command is virtualized so that the VMkernel can isolate the LUN to the owning virtual machine. Otherwise, all physical characteristics of the underlying hardware are exposed. Physical mode is useful to run SAN management agents or other SCSI target-based software in the virtual machine. Physical mode also allows virtual-to-physical clustering for cost-effective high availability.

VMFS5 supports greater than 2TB disk size for RDMs in physical compatibility mode only. The following restrictions apply:

- You cannot relocate larger than 2TB RDMs to datastores other than VMFS5.
- You cannot convert larger than 2TB RDMs to virtual disks, or perform other operations that involve RDM to virtual disk conversion. Such operations include Storage vMotion, migration, and cloning.

Dynamic Name Resolution

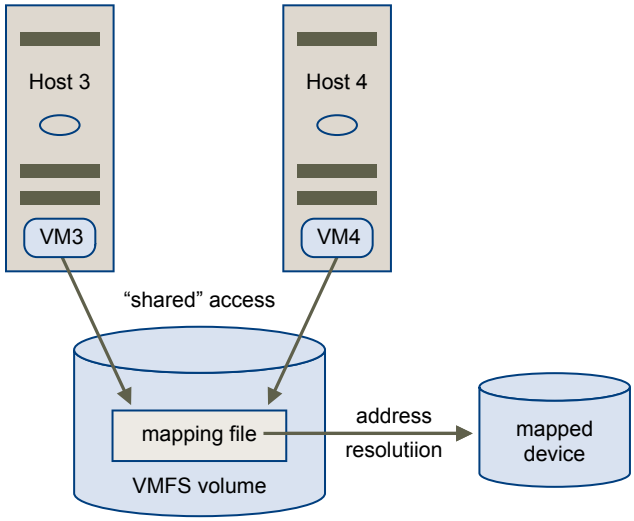
The RDM file supports dynamic name resolution when a path to a raw device changes.

VMFS uniquely identifies all mapped storage devices, and the identification is stored in its internal data structures. Any change in the path to a raw device, such as a Fibre Channel switch failure or the addition of a new HBA, can change the device name. Dynamic name resolution resolves these changes and automatically associates the original device with its new name.

Raw Device Mapping with Virtual Machine Clusters

Use an RDM with virtual machine clusters that need to access the same raw LUN for failover scenarios. The setup is similar to that of a virtual machine cluster that accesses the same virtual disk file, but an RDM replaces the virtual disk file.

Figure 14-3. Access from Clustered Virtual Machines



Comparing Available SCSI Device Access Modes

The ways of accessing a SCSI-based storage device include a virtual disk file on a VMFS datastore, virtual mode RDM, and physical mode RDM.

To help you choose among the available access modes for SCSI devices, the following table provides a quick comparison of features available with the different modes.

Table 14-1. Features Available with Virtual Disks and Raw Device Mappings

ESXi Features	Virtual Disk File	Virtual Mode RDM	Physical Mode RDM
SCSI Commands Passed Through	No	No	Yes REPORT LUNs is not passed through
vCenter Server Support	Yes	Yes	Yes
Snapshots	Yes	Yes	No
Distributed Locking	Yes	Yes	Yes
Clustering	Cluster-in-a-box only	Cluster-in-a-box cluster-across-boxes	Physical-to-virtual clustering cluster-across-boxes
SCSI Target-Based Software	No	No	Yes

VMware recommends that you use virtual disk files for the cluster-in-a-box type of clustering. If you plan to reconfigure your cluster-in-a-box clusters as cluster-across-boxes clusters, use virtual mode RDMs for the cluster-in-a-box clusters.

Create Virtual Machines with RDMs

When you give your virtual machine direct access to a raw SAN LUN, you create a mapping file (RDM) that resides on a VMFS datastore and points to the LUN. Although the mapping file has the same `.vmdk` extension as a regular virtual disk file, the RDM file contains only mapping information. The actual virtual disk data is stored directly on the LUN.

You can create the RDM as an initial disk for a new virtual machine or add it to an existing virtual machine. When creating the RDM, you specify the LUN to be mapped and the datastore on which to put the RDM.

Procedure

1 Follow all steps required to create a custom virtual machine.

2 In the Select a Disk page, select **Raw Device Mapping**, and click **Next**.

3 From the list of SAN disks or LUNs, select a raw LUN for your virtual machine to access directly.

4 Select a datastore for the RDM mapping file.

You can place the RDM file on the same datastore where your virtual machine configuration file resides, or select a different datastore.

NOTE To use vMotion for virtual machines with enabled NPIV, make sure that the RDM files of the virtual machines are located on the same datastore. You cannot perform Storage vMotion when NPIV is enabled.

5 Select a compatibility mode.

Option	Description
Physical	Allows the guest operating system to access the hardware directly. Physical compatibility is useful if you are using SAN-aware applications on the virtual machine. However, powered on virtual machines that use RDMs configured for physical compatibility cannot be migrated if the migration involves copying the disk. Such virtual machines cannot be cloned or cloned to a template either.
Virtual	Allows the RDM to behave as if it were a virtual disk, so you can use such features as snapshotting, cloning, and so on.

6 Select a virtual device node.

7 If you select Independent mode, choose one of the following.

Option	Description
Persistent	Changes are immediately and permanently written to the disk.
Nonpersistent	Changes to the disk are discarded when you power off or revert to the snapshot.

8 Click **Next**.

9 In the Ready to Complete New Virtual Machine page, review your selections.

10 Click **Finish** to complete your virtual machine.

Manage Paths for a Mapped Raw LUN

You can manage paths for mapped raw LUNs.

Procedure

1 Log in as administrator or as the owner of the virtual machine to which the mapped disk belongs.

2 Select the virtual machine from the Inventory panel.

3 On the **Summary** tab, click **Edit Settings**.

4 On the **Hardware** tab, select **Hard Disk**, then click **Manage Paths**.

5 Use the Manage Paths dialog box to enable or disable your paths, set multipathing policy, and specify the preferred path.

For information on managing paths, see Chapter 17, “Understanding Multipathing and Failover,” on page 153.

Solid State Disks Enablement 15

In addition to regular hard disk drives, ESXi supports Solid State Disks (SSDs).

Unlike the regular hard disks that are electromechanical devices containing moving parts, SSDs use semiconductors as their storage medium and have no moving parts.

On several storage arrays, the ESXi host can automatically distinguish SSDs from traditional hard disks. To tag the SSD devices that are not detected automatically, you can use PSA SATP claim rules.

This chapter includes the following topics:

- "Benefits of SSD Enablement," on page 143
- "Auto-Detection of SSD Devices," on page 143
- "Tag Devices as SSD," on page 144
- "Untag an SSD Device," on page 145
- "Untag an Automatically Detected SSD Device," on page 146
- "Tag Devices as Local," on page 146
- "Identify SSD Devices," on page 147
- "Identifying a Virtual SSD Device," on page 148

Benefits of SSD Enablement

SSDs are very resilient and provide faster access to data.

SSD enablement results in several benefits:

- It enables usage of SSD as swap space for improved system performance. For information about using SSD datastores to allocate space for host cache, see the *vSphere Resource Management* documentation.
- It increases virtual machine consolidation ratio as SSDs can provide very high I/O throughput.
- It supports identification of virtual SSD device by the guest operating system.

Auto-Detection of SSD Devices

ESXi enables automatic detection of SSD devices by using an inquiry mechanism based on T10 standards.

ESXi enables detection of the SSD devices on a number of storage arrays. Check with your vendor whether your storage array supports ESXi SSD device detection.

You can use PSA SATP claim rules to tag devices that cannot be auto-detected.

Tag Devices as SSD

You can use PSA SATP claim rules to tag SSD devices that are not detected automatically.

Only devices that are consumed by the PSA Native Multipathing (NMP) plugin can be tagged.

If you need more information about the commands listed in this topic, see the *Getting Started with vSphere Command-Line Interfaces* and *vSphere Command-Line Interface Concepts and Examples* documentation.

Procedure

1 Identify the device to be tagged and its SATP.

```
esxcli storage nmp device list
```

The command results in the following information.

```
naa.6006016015301d00167ce6e2ddb3de11
Device Display Name: DGC Fibre Channel Disk (naa.6006016015301d00167ce6e2ddb3de11)
Storage Array Type: VMW_SATP_CX
Storage Array Type Device Config: {navireg ipfilter}
Path Selection Policy: VMW_PSP_MRU
Path Selection Policy Device Config: Current Path=vmhba4:C0:T0:L25
Working Paths: vmhba4:C0:T0:L25
```

2 Note down the SATP associated with the device.

3 Add a PSA claim rule to mark the device as SSD.

- You can add a claim rule by specifying the device name.

  ```
  esxcli storage nmp satp rule add –s SATP –-device device_name –-option=enable_ssd
  ```

- You can add a claim rule by specifying the vendor name and the model name.

  ```
  esxcli storage nmp satp rule add –s SATP –V vendor_name –M model_name –-option=enable_ssd
  ```

- You can add a claim rule based on the transport protocol.

  ```
  esxcli storage nmp satp rule add –s SATP –-transport transport_protocol –-
  option=enable_ssd
  ```

- You can add a claim rule based on the driver name.

  ```
  esxcli storage nmp satp rule add –s SATP –-driver driver_name –-option=enable_ssd
  ```

4 Unclaim the device.

- You can unclaim the device by specifying the device name.

  ```
  esxcli storage core claiming unclaim –-type device –-device device_name
  ```

- You can unclaim the device by specifying the vendor name and the model name.

  ```
  esxcli storage core claiming unclaim –-type device –V vendor_name –M model_name
  ```

- You can unclaim the device based on the transport protocol.

  ```
  esxcli storage core claiming unclaim –-type device –-transport transport_protocol
  ```

- You can unclaim the device based on the driver name.

  ```
  esxcli storage core claiming unclaim –-type device –-driver driver_name
  ```

5 Reclaim the device by running the following commands.

```
esxcli storage core claimrule load
esxcli storage core claimrule run
```

6 Verify if devices are tagged as SSD.

```
esxcli storage core device list –d device_name
```

The command output indicates if a listed device is tagged as SSD.

```
Is SSD: true
```

What to do next

If the SSD device that you want to tag is shared among multiple hosts, make sure that you tag the device from all the hosts that share the device.

Untag an SSD Device

You can untag a device tagged as SSD whenever required.

This topic lists commands that you need to use to untag an SSD device. For more information about these commands, see the *Getting Started with vSphere Command-Line Interfaces* and *vSphere Command-Line Interface Concepts and Examples* documentation.

Procedure

1 Identify the device that you want to untag.

```
esxcli storage nmp device list
```

The command results in the following information.

```
naa.6006016015301d00167ce6e2ddb3de11
Device Display Name: DGC Fibre Channel Disk (naa.6006016015301d00167ce6e2ddb3de11)
Storage Array Type: VMW_SATP_CX
Storage Array Type Device Config: {navireg ipfilter}
Path Selection Policy: VMW_PSP_MRU
Path Selection Policy Device Config: Current Path=vmhba4:C0:T0:L25
Working Paths: vmhba4:C0:T0:L25
```

2 Note down the SATP associated with the device.

3 Remove the claim rule that was used to tag the device.

```
esxcli storage nmp satp rule remove –s SATP ––device device_name
```

For example,

```
esxcli storage nmp satp rule remove –s VMW_SATP_CX ––device naa.
6006016042fa19010a12d9b16d6ade11
```

4 Unclaim the device.

```
esxcli storage core claiming unclaim ––type device ––device naa.
6006016042fa19010a12d9b16d6ade11
```

5 Reclaim the device by running the following commands:

```
esxcli storage core claimrule load
esxcli storage core claimrule run
```

6 Check the device status by running the following command.

```
esxcli storage core device list –d device_name
```

The command output indicates whether the disk is untagged.

```
Is SSD: false
```

Untag an Automatically Detected SSD Device

You can tag an automatically detected SSD device as a non-SSD device.

Procedure

1 Identify the device that you want to untag.

```
esxcli storage nmp device list
```

The command results in the following information.

```
naa.6006016015301d00167ce6e2ddb3de11
Device Display Name: DGC Fibre Channel Disk (naa.6006016015301d00167ce6e2ddb3de11)
Storage Array Type: VMW_SATP_CX
Storage Array Type Device Config: {navireg ipfilter}
Path Selection Policy: VMW_PSP_MRU
Path Selection Policy Device Config: Current Path=vmhba4:C0:T0:L25
Working Paths: vmhba4:C0:T0:L25
```

2 Note down the SATP associated with the device.

3 Add a claim rule with the option *disable_ssd*.

```
esxcli storage nmp satp rule add –s SATP ––device device_name ––option disable_ssd
```

For example,

```
esxcli storage nmp satp rule add –s VMW_SATP_CX ––device naa.6006016042fa19010a12d9b16d6ade11
––option disable_ssd
```

4 Unclaim the device.

```
esxcli storage core claiming unclaim ––type device ––device naa.
6006016042fa19010a12d9b16d6ade11
```

5 Reclaim the device by running the following commands:

```
esxcli storage core claimrule load
esxcli storage core claimrule run
```

6 Check the device status by running the following command.

```
esxcli storage core device list –d device_name
```

The command output indicates whether the disk is untagged.

```
Is SSD: false
```

Tag Devices as Local

ESXi enables you to tag devices as local. This is useful in cases where you are unable to determine whether certain SAS controllers are local or remote. This ability to tag devices can also be used in Stateless ESXi system configurations to autoformat local devices.

For more information about the commands listed in this topic, see the *Getting Started with vSphere Command-Line Interfaces* and *vSphere Command-Line Interface Concepts and Examples* documentation.

Prerequisites

Make sure that the device is claimed by VMW_SATP_LOCAL.

Procedure

1 Identify the device to be tagged and its SATP.

```
esxcli storage nmp device list
```

The command results in the following information.

```
naa.6006016015301d00167ce6e2ddb3de11
Device Display Name: DGC Fibre Channel Disk (naa.6006016015301d00167ce6e2ddb3de11)
Storage Array Type: VMW_SATP_CX
Storage Array Type Device Config: {navireg ipfilter}
Path Selection Policy: VMW_PSP_MRU
Path Selection Policy Device Config: Current Path=vmhba4:C0:T0:L25
Working Paths: vmhba4:C0:T0:L25
```

2 Note down the SATP associated with the device.

3 Add a PSA claim rule.

```
esxcli storage nmp satp rule add –s SATP ––device device_name option="enable_local"
```

For example,

```
esxcli storage nmp satp rule add –s VMW_SATP_LOCAL ––device naa.
6006016042fa19010a12d9b16d6ade11 option="enable_local"
```

4 Unclaim the device.

```
esxcli storage core claiming unclaim ––type device ––device naa.
6006016042fa19010a12d9b16d6ade11
```

5 Reclaim the device by running the following commands.

```
esxcli storage core claimrule load
esxcli storage core claimrule run
```

6 Check the status by running the following command.

```
esxcli storage core device list –d device_name
```

The command output indicates whether the disk is remote or local.

Identify SSD Devices

You can identify the SSD devices in your storage network.

Prerequisites

Before you identify an SSD device, ensure that the device is tagged as SSD.

Procedure

1 List the devices.

```
esxcli storage core device list
```

The command output includes the following information about the listed device.

```
Is SSD: true
```

2 Verify whether the value of the flag `Is SSD` is `true`.

Identifying a Virtual SSD Device

ESXi allows operating systems to auto-detect VMDKs residing on SSD datastores as SSD devices.

To verify if this feature is enabled, guest operating systems can use standard inquiry commands such as SCSI VPD Page (B1h) for SCSI devices and ATA IDENTIFY DEVICE (Word 217) for IDE devices.

For linked clones, native snapshots, and delta-disks, the inquiry commands report the virtual SSD status of the base disk.

Operating systems can auto-detect a VMDK as SSD under the following conditions:

- Detection of virtual SSDs is supported on ESXi 5 hosts and Virtual Hardware version 8.
- Detection of virtual SSDs is supported only on VMFS5 or later.
- If VMDKs are located on shared VMFS datastores with SSD device extents, the device must be marked as SSD on all hosts.
- For a VMDK to be detected as virtual SSD, all underlying physical extents should be SSD-backed.

VMkernel and Storage 16

The VMkernel is a high-performance operating system that runs directly on the ESXi host. The VMkernel manages most of the physical resources on the hardware, including memory, physical processors, storage, and networking controllers.

To manage storage, VMkernel has a storage subsystem that supports several Host Bus Adapters (HBAs) including parallel SCSI, SAS, Fibre Channel, FCoE, and iSCSI. These HBAs connect a wide variety of active-active, active-passive, and ALUA storage arrays that are certified for use with the VMkernel. See the *vSphere Compatibility Guide* for a list of the supported HBAs and storage arrays.

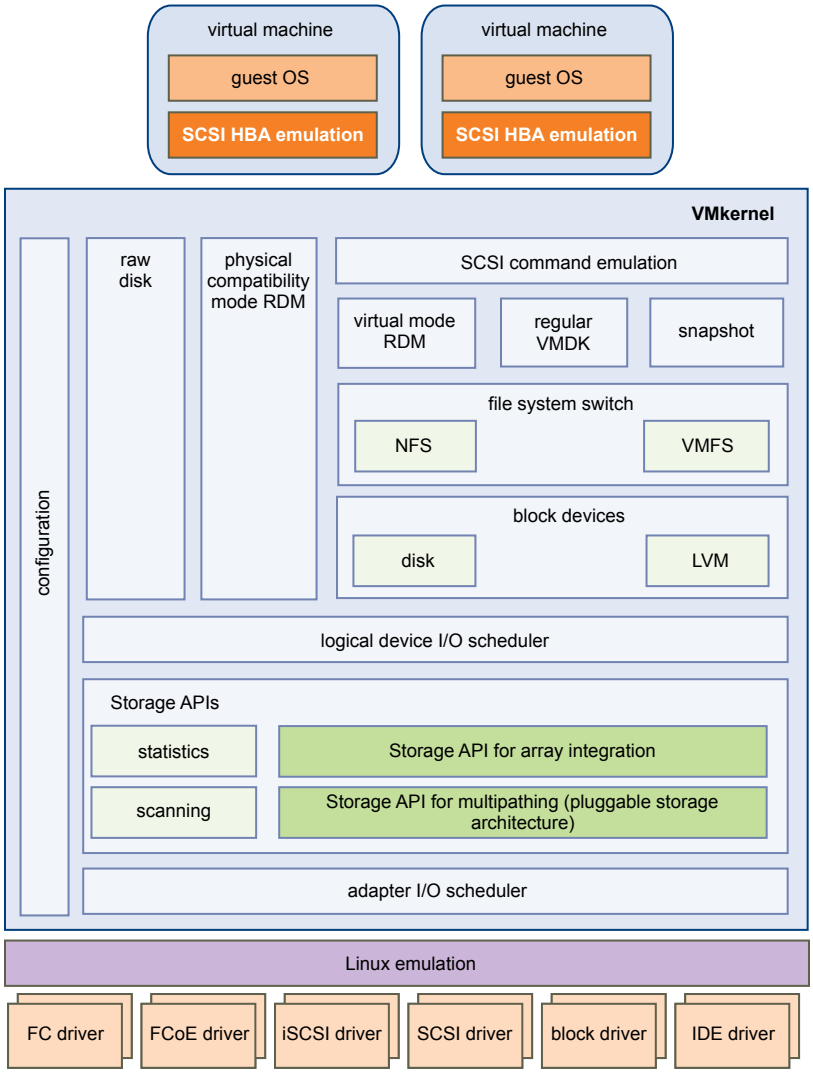
The primary file system that the VMkernel uses is the VMware Virtual Machine File System (VMFS). VMFS is a cluster file system designed and optimized to support large files such as virtual disks and swap files. The VMkernel also supports the storage of virtual disks on NFS file systems.

The storage I/O path provides virtual machines with access to storage devices through device emulation. This device emulation allows a virtual machine to access files on a VMFS or NFS file system as if they were SCSI devices. The VMkernel provides storage virtualization functions such as the scheduling of I/O requests from multiple virtual machines and multipathing.

In addition, VMkernel offers several Storage APIs that enable storage partners to integrate and optimize their products for vSphere.

The following graphic illustrates the basics of the VMkernel core, with special attention to the storage stack. Storage-related modules reside between the logical device I/O scheduler and the adapter I/O scheduler layers.

Figure 16-1. VMkernel and Storage



Storage APIs

Storage APIs is a family of APIs used by third-party hardware, software, and storage providers to develop components that enhance several vSphere features and solutions.

This publication describes the following sets of Storage APIs and explains how they contribute to your storage environment. For information about other APIs from this family, including Storage API - Data Protection and Storage API - Site Recovery Manager, see the VMware Web site.

- Storage APIs - Multipathing, also known as the Pluggable Storage Architecture (PSA). PSA is a collection of VMkernel APIs that allows storage partners to enable and certify their arrays asynchronous to ESXi release schedules, as well as deliver performance-enhancing, multipathing and load-balancing behaviors that are optimized for each array. For more information, see "Managing Multiple Paths," on page 158.
- Storage APIs - Array Integration, formerly known as VAAI, include the following APIs:
  - Hardware Acceleration APIs. Allows arrays to integrate with vSphere to transparently offload certain storage operations to the array. This integration significantly reduces CPU overhead on the host. See Chapter 18, "Storage Hardware Acceleration," on page 173.

- Array Thin Provisioning APIs. Help to monitor space use on thin-provisioned storage arrays to prevent out-of-space conditions, and to perform space reclamation. See "Array Thin Provisioning and VMFS Datastores," on page 186.

- Storage APIs - Storage Awareness. These vCenter Server-based APIs enable storage arrays to inform the vCenter Server about their configurations, capabilities, and storage health and events. See Chapter 20, "Using Storage Vendor Providers," on page 191.

Understanding Multipathing and Failover

17

To maintain a constant connection between a host and its storage, ESXi supports multipathing. Multipathing is a technique that lets you use more than one physical path that transfers data between the host and an external storage device.

In case of a failure of any element in the SAN network, such as an adapter, switch, or cable, ESXi can switch to another physical path, which does not use the failed component. This process of path switching to avoid failed components is known as path failover.

In addition to path failover, multipathing provides load balancing. Load balancing is the process of distributing I/O loads across multiple physical paths. Load balancing reduces or removes potential bottlenecks.

NOTE Virtual machine I/O might be delayed for up to sixty seconds while path failover takes place. These delays allow the SAN to stabilize its configuration after topology changes. In general, the I/O delays might be longer on active-passive arrays and shorter on active-active arrays.

This chapter includes the following topics:

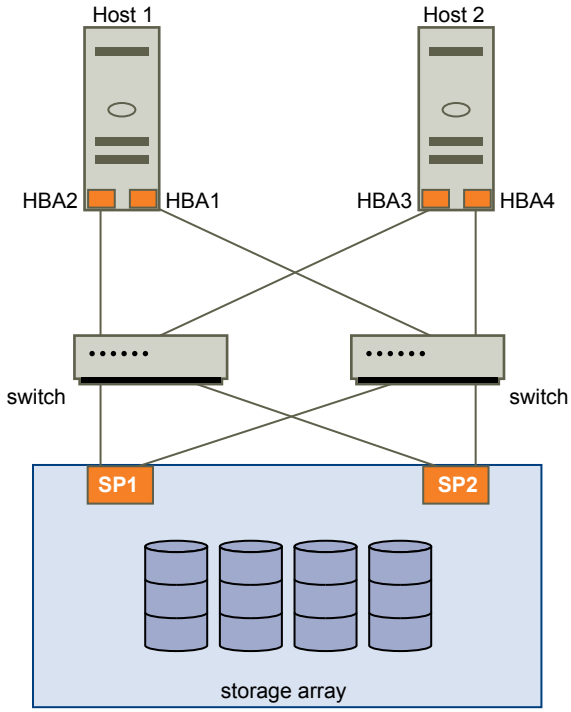
- "Failover with Fibre Channel," on page 153
- "Host-Based Failover with iSCSI," on page 154
- "Array-Based Failover with iSCSI," on page 156
- "Path Failover and Virtual Machines," on page 157
- "Managing Multiple Paths," on page 158
- "VMware Multipathing Module," on page 159
- "Path Scanning and Claiming," on page 161
- "Managing Storage Paths and Multipathing Plug-Ins," on page 164

Failover with Fibre Channel

To support multipathing, your host typically has two or more HBAs available. This configuration supplements the SAN multipathing configuration that generally provides one or more switches in the SAN fabric and one or more storage processors on the storage array device itself.

In the following illustration, multiple physical paths connect each server with the storage device. For example, if HBA1 or the link between HBA1 and the FC switch fails, HBA2 takes over and provides the connection between the server and the switch. The process of one HBA taking over for another is called HBA failover.

Figure 17-1. Multipathing and Failover with Fibre Channel



Similarly, if SP1 fails or the links between SP1 and the switches breaks, SP2 takes over and provides the connection between the switch and the storage device. This process is called SP failover. VMware ESXi supports both HBA and SP failovers with its multipathing capability.

Host-Based Failover with iSCSI

When setting up your ESXi host for multipathing and failover, you can use multiple iSCSI HBAs or multiple NICs depending on the type of iSCSI adapters on your host.

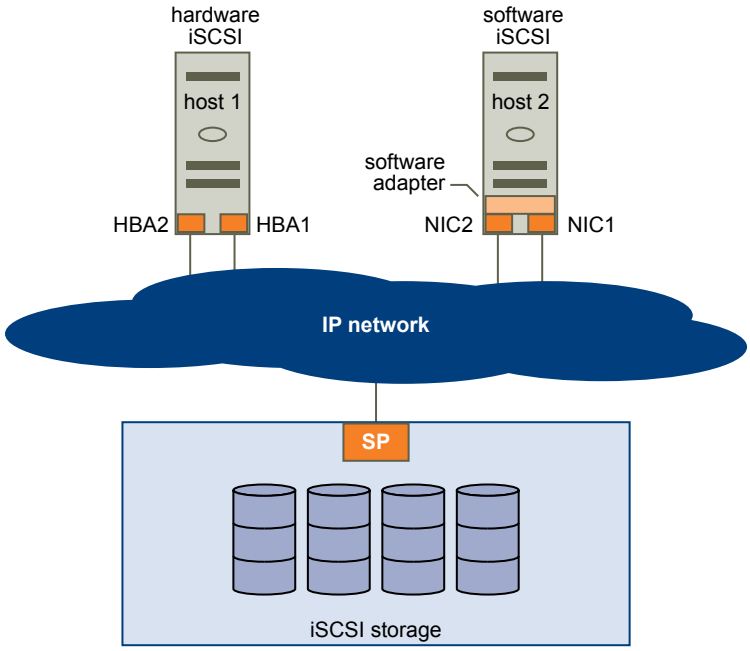
For information on different types of iSCSI adapters, see "iSCSI Initiators," on page 63.

When you use multipathing, specific considerations apply.

- ESXi does not support multipathing when you combine an independent hardware adapter with software iSCSI or dependent iSCSI adapters in the same host.
- Multipathing between software and dependent adapters within the same host is supported.
- On different hosts, you can mix both dependent and independent adapters.

The following illustration shows multipathing setups possible with different types of iSCSI initiators.

Figure 17-2. Host-Based Path Failover



Failover with Hardware iSCSI

With hardware iSCSI, the host typically has two or more hardware iSCSI adapters available, from which the storage system can be reached using one or more switches. Alternatively, the setup might include one adapter and two storage processors so that the adapter can use a different path to reach the storage system.

On the Host-Based Path Failover illustration, Host1 has two hardware iSCSI adapters, HBA1 and HBA2, that provide two physical paths to the storage system. Multipathing plug-ins on your host, whether the VMkernel NMP or any third-party MPPs, have access to the paths by default and can monitor health of each physical path. If, for example, HBA1 or the link between HBA1 and the network fails, the multipathing plug-ins can switch the path over to HBA2.

Failover with Software iSCSI

With software iSCSI, as shown on Host 2 of the Host-Based Path Failover illustration, you can use multiple NICs that provide failover and load balancing capabilities for iSCSI connections between your host and storage systems.

For this setup, because multipathing plug-ins do not have direct access to physical NICs on your host, you first need to connect each physical NIC to a separate VMkernel port. You then associate all VMkernel ports with the software iSCSI initiator using a port binding technique. As a result, each VMkernel port connected to a separate NIC becomes a different path that the iSCSI storage stack and its storage-aware multipathing plug-ins can use.

For information on how to configure multipathing for software iSCSI, see "Setting Up iSCSI Network," on page 74.

Array-Based Failover with iSCSI

Some iSCSI storage systems manage path use of their ports automatically and transparently to ESXi.

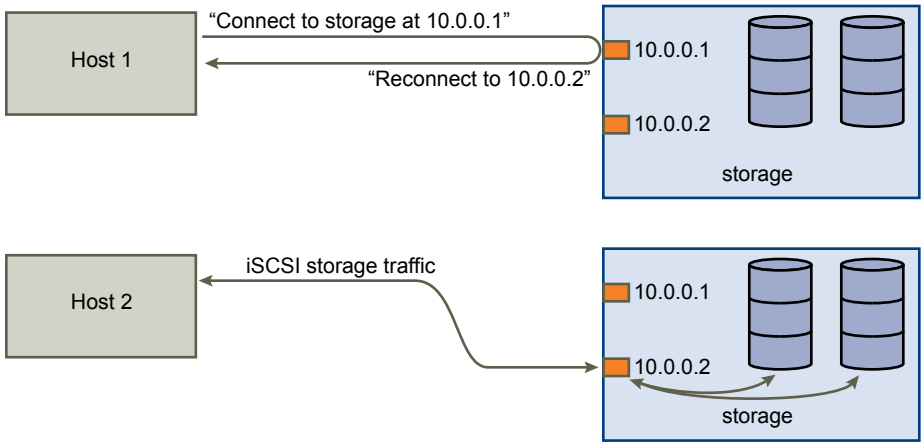
When using one of these storage systems, your host does not see multiple ports on the storage and cannot choose the storage port it connects to. These systems have a single virtual port address that your host uses to initially communicate. During this initial communication, the storage system can redirect the host to communicate with another port on the storage system. The iSCSI initiators in the host obey this reconnection request and connect with a different port on the system. The storage system uses this technique to spread the load across available ports.

If the ESXi host loses connection to one of these ports, it automatically attempts to reconnect with the virtual port of the storage system, and should be redirected to an active, usable port. This reconnection and redirection happens quickly and generally does not disrupt running virtual machines. These storage systems can also request that iSCSI initiators reconnect to the system, to change which storage port they are connected to. This allows the most effective use of the multiple ports.

The Port Redirection illustration shows an example of port redirection. The host attempts to connect to the 10.0.0.1 virtual port. The storage system redirects this request to 10.0.0.2. The host connects with 10.0.0.2 and uses this port for I/O communication.

NOTE The storage system does not always redirect connections. The port at 10.0.0.1 could be used for traffic, also.

Figure 17-3. Port Redirection



If the port on the storage system that is acting as the virtual port becomes unavailable, the storage system reassigns the address of the virtual port to another port on the system. Port Reassignment shows an example of this type of port reassignment. In this case, the virtual port 10.0.0.1 becomes unavailable and the storage system reassigns the virtual port IP address to a different port. The second port responds to both addresses.

Figure 17-4. Port Reassignment

With this form of array-based failover, you can have multiple paths to the storage only if you use multiple ports on the ESXi host. These paths are active-active. For additional information, see "iSCSI Session Management," on page 87.

Path Failover and Virtual Machines

Path failover occurs when the active path to a LUN is changed from one path to another, usually because of a SAN component failure along the current path.

When a path fails, storage I/O might pause for 30 to 60 seconds until your host determines that the link is unavailable and completes failover. If you attempt to display the host, its storage devices, or its adapters, the operation might appear to stall. Virtual machines with their disks installed on the SAN can appear unresponsive. After failover is complete, I/O resumes normally and the virtual machines continue to run.

However, when failovers take a long time to complete, a Windows virtual machine might interrupt the I/O and eventually fail. To avoid the failure, set the disk timeout value for the Windows virtual machine to at least 60 seconds.

Set Timeout on Windows Guest OS

Increase the standard disk timeout value on a Windows guest operating system to avoid disruptions during a path failover.

This procedure explains how to change the timeout value by using the Windows registry.

Prerequisites

Back up the Windows registry.

Procedure

1 Select **Start > Run**.

2 Type **`regedit.exe`**, and click **OK**.

3 In the left-panel hierarchy view, double-click **HKEY_LOCAL_MACHINE > System > CurrentControlSet > Services > Disk**.

4 Double-click **TimeOutValue**.

5 Set the value data to 0x3c (hexadecimal) or 60 (decimal) and click **OK**.

After you make this change, Windows waits at least 60 seconds for delayed disk operations to complete before it generates errors.

6 Reboot guest OS for the change to take effect.

Managing Multiple Paths

To manage storage multipathing, ESXi uses a collection of Storage APIs, also called the Pluggable Storage Architecture (PSA). The PSA is an open, modular framework that coordinates the simultaneous operation of multiple multipathing plug-ins (MPPs). The PSA allows 3rd party software developers to design their own load balancing techniques and failover mechanisms for particular storage array, and insert their code directly into the ESXi storage I/O path.

Topics discussing path management use the following acronyms.

Table 17-1. Multipathing Acronyms

Acronym	Definition
PSA	Pluggable Storage Architecture
NMP	Native Multipathing Plug-In. Generic VMware multipathing module.
PSP	Path Selection Plug-In, also called Path Selection Policy. Handles path selection for a given device.
SATP	Storage Array Type Plug-In, also called Storage Array Type Policy. Handles path failover for a given storage array.

The VMkernel multipathing plug-in that ESXi provides by default is the VMware Native Multipathing Plug-In (NMP). The NMP is an extensible module that manages sub plug-ins. There are two types of NMP sub plug-ins, Storage Array Type Plug-Ins (SATPs), and Path Selection Plug-Ins (PSPs). SATPs and PSPs can be built-in and provided by VMware, or can be provided by a third party.

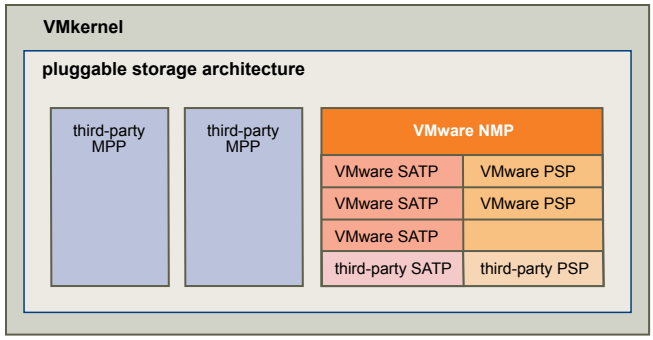
If more multipathing functionality is required, a third party can also provide an MPP to run in addition to, or as a replacement for, the default NMP.

When coordinating the VMware NMP and any installed third-party MPPs, the PSA performs the following tasks:

- Loads and unloads multipathing plug-ins.
- Hides virtual machine specifics from a particular plug-in.
- Routes I/O requests for a specific logical device to the MPP managing that device.
- Handles I/O queueing to the logical devices.
- Implements logical device bandwidth sharing between virtual machines.
- Handles I/O queueing to the physical storage HBAs.
- Handles physical path discovery and removal.
- Provides logical device and physical path I/O statistics.

As the Pluggable Storage Architecture illustration shows, multiple third-party MPPs can run in parallel with the VMware NMP. When installed, the third-party MPPs replace the behavior of the NMP and take complete control of the path failover and the load-balancing operations for specified storage devices.

Figure 17-5. Pluggable Storage Architecture



The multipathing modules perform the following operations:

- Manage physical path claiming and unclaiming.
- Manage creation, registration, and deregistration of logical devices.
- Associate physical paths with logical devices.
- Support path failure detection and remediation.
- Process I/O requests to logical devices:
  - Select an optimal physical path for the request.
  - Depending on a storage device, perform specific actions necessary to handle path failures and I/O command retries.
- Support management tasks, such as reset of logical devices.

VMware Multipathing Module

By default, ESXi provides an extensible multipathing module called the Native Multipathing Plug-In (NMP).

Generally, the VMware NMP supports all storage arrays listed on the VMware storage HCL and provides a default path selection algorithm based on the array type. The NMP associates a set of physical paths with a specific storage device, or LUN. The specific details of handling path failover for a given storage array are delegated to a Storage Array Type Plug-In (SATP). The specific details for determining which physical path is used to issue an I/O request to a storage device are handled by a Path Selection Plug-In (PSP). SATPs and PSPs are sub plug-ins within the NMP module.

With ESXi, the appropriate SATP for an array you use will be installed automatically. You do not need to obtain or download any SATPs.

VMware SATPs

Storage Array Type Plug-Ins (SATPs) run in conjunction with the VMware NMP and are responsible for array-specific operations.

ESXi offers a SATP for every type of array that VMware supports. It also provides default SATPs that support non-specific active-active and ALUA storage arrays, and the local SATP for direct-attached devices. Each SATP accommodates special characteristics of a certain class of storage arrays and can perform the array-specific operations required to detect path state and to activate an inactive path. As a result, the NMP module itself can work with multiple storage arrays without having to be aware of the storage device specifics.

After the NMP determines which SATP to use for a specific storage device and associates the SATP with the physical paths for that storage device, the SATP implements the tasks that include the following:

- Monitors the health of each physical path.
- Reports changes in the state of each physical path.

- Performs array-specific actions necessary for storage fail-over. For example, for active-passive devices, it can activate passive paths.

VMware PSPs

Path Selection Plug-Ins (PSPs) are sub plug-ins of the VMware NMP and are responsible for choosing a physical path for I/O requests.

The VMware NMP assigns a default PSP for each logical device based on the SATP associated with the physical paths for that device. You can override the default PSP. For information, see "Change the Path Selection Policy," on page 163.

By default, the VMware NMP supports the following PSPs:

VMW_PSP_MRU

The host selects the path that it used most recently. When the path becomes unavailable, the host selects an alternative path. The host does not revert back to the original path when that path becomes available again. There is no preferred path setting with the MRU policy. MRU is the default policy for most active-passive storage devices.

Displayed in the vSphere Client as the Most Recently Used (VMware) path selection policy.

VMW_PSP_FIXED

The host uses the designated preferred path, if it has been configured. Otherwise, it selects the first working path discovered at system boot time. If you want the host to use a particular preferred path, specify it through the vSphere Client. Fixed is the default policy for most active-active storage devices.

Displayed in the vSphere Client as the Fixed (VMware) path selection policy.

VMW_PSP_RR

The host uses an automatic path selection algorithm rotating through all active paths when connecting to active-passive arrays, or through all available paths when connecting to active-active arrays. RR is the default for a number of arrays and can be used with both active-active and active-passive arrays to implement load balancing across paths for different LUNs.

Displayed in the vSphere Client as the Round Robin (VMware) path selection policy.

VMware NMP Flow of I/O

When a virtual machine issues an I/O request to a storage device managed by the NMP, the following process takes place.

1 The NMP calls the PSP assigned to this storage device.

2 The PSP selects an appropriate physical path on which to issue the I/O.

3 The NMP issues the I/O request on the path selected by the PSP.

4 If the I/O operation is successful, the NMP reports its completion.

5 If the I/O operation reports an error, the NMP calls the appropriate SATP.

6 The SATP interprets the I/O command errors and, when appropriate, activates the inactive paths.

7 The PSP is called to select a new path on which to issue the I/O.

Path Scanning and Claiming

When you start your ESXi host or rescan your storage adapter, the host discovers all physical paths to storage devices available to the host. Based on a set of claim rules, the host determines which multipathing plug-in (MPP) should claim the paths to a particular device and become responsible for managing the multipathing support for the device.

By default, the host performs a periodic path evaluation every 5 minutes causing any unclaimed paths to be claimed by the appropriate MPP.

The claim rules are numbered. For each physical path, the host runs through the claim rules starting with the lowest number first. The attributes of the physical path are compared to the path specification in the claim rule. If there is a match, the host assigns the MPP specified in the claim rule to manage the physical path. This continues until all physical paths are claimed by corresponding MPPs, either third-party multipathing plug-ins or the native multipathing plug-in (NMP).

For the paths managed by the NMP module, a second set of claim rules is applied. These rules determine which Storage Array Type Plug-In (SATP) should be used to manage the paths for a specific array type, and which Path Selection Plug-In (PSP) is to be used for each storage device.

Use the vSphere Client to view which SATP and PSP the host is using for a specific storage device and the status of all available paths for this storage device. If needed, you can change the default VMware PSP using the vSphere Client. To change the default SATP, you need to modify claim rules using the vSphere CLI.

You can find some information about modifying claim rules in "Managing Storage Paths and Multipathing Plug-Ins," on page 164.

For more information about the commands available to manage PSA, see *Getting Started with vSphere Command-Line Interfaces*.

For a complete list of storage arrays and corresponding SATPs and PSPs, see the SAN Array Model Reference section of the *vSphere Compatibility Guide*.

Viewing the Paths Information

Use the vSphere Client to determine which SATP and PSP the ESXi host uses for a specific storage device and the status of all available paths for this storage device. You can access the path information from both the Datastores and Devices views. For datastores, you review the paths that connect to the device the datastore is deployed on.

The path information includes the SATP assigned to manage the device, the path selection policy (PSP), a list of paths, and the status of each path. The following path status information can appear:

Active — Paths available for issuing I/O to a LUN. A single or multiple working paths currently used for transferring data are marked as Active (I/O).

NOTE For hosts that run ESX/ESXi version 3.5 or earlier, the term active means the only path that the host is using to issue I/O to a LUN.

Standby — If active paths fail, the path can quickly become operational and can be used for I/O.

Disabled — The path is disabled and no data can be transferred.

Dead — The software cannot connect to the disk through this path.

If you are using the **Fixed** path policy, you can see which path is the preferred path. The preferred path is marked with an asterisk (*) in the Preferred column.

For each path you can also display the path's name. The name includes parameters that describe the path: adapter ID, target ID, and device ID. Usually, the path's name has the format similar to the following:

```
fc.adapterID-fc.targetID-naa.deviceID
```

NOTE When you use the host profiles editor to edit paths, you must specify all three parameters that describe a path, adapter ID, target ID, and device ID.

View Datastore Paths

Use the vSphere Client to review the paths that connect to storage devices the datastores are deployed on.

Procedure

1 Log in to the vSphere Client and select a host from the inventory panel.

2 Click the **Configuration** tab and click **Storage** in the Hardware panel.

3 Click **Datastores** under View.

4 From the list of configured datastores, select the datastore whose paths you want to view, and click **Properties**.

5 Under Extents, select the storage device whose paths you want to view and click **Manage Paths**.

6 In the Paths panel, select the path to view.

The panel underneath displays the path's name. The name includes parameters describing the path: adapter ID, target ID, and device ID.

7 (Optional) To extract the path's parameters, right-click the path and select **Copy path to clipboard**.

View Storage Device Paths

Use the vSphere Client to view which SATP and PSP the host uses for a specific storage device and the status of all available paths for this storage device.

Procedure

1 Log in to the vSphere Client and select a server from the inventory panel.

2 Click the **Configuration** tab and click **Storage** in the Hardware panel.

3 Click **Devices** under View.

4 Select the storage device whose paths you want to view and click **Manage Paths**.

5 In the Paths panel, select the path to view.

The panel underneath displays the path's name. The name includes parameters describing the path: adapter ID, target ID, and device ID.

6 (Optional) To extract the path's parameters, right-click the path and select **Copy path to clipboard**.

Setting a Path Selection Policy

For each storage device, the ESXi host sets the path selection policy based on the claim rules.

By default, VMware supports the following path selection policies. If you have a third-party PSP installed on your host, its policy also appears on the list.

Fixed (VMware)	The host uses the designated preferred path, if it has been configured. Otherwise, it selects the first working path discovered at system boot time. If you want the host to use a particular preferred path, specify it through the vSphere Client. Fixed is the default policy for most active-active storage devices.
Most Recently Used (VMware)	The host selects the path that it used most recently. When the path becomes unavailable, the host selects an alternative path. The host does not revert back to the original path when that path becomes available again. There is no preferred path setting with the MRU policy. MRU is the default policy for most active-passive storage devices.
Round Robin (VMware)	The host uses an automatic path selection algorithm rotating through all active paths when connecting to active-passive arrays, or through all available paths when connecting to active-active arrays. RR is the default for a number of arrays and can be used with both active-active and active-passive arrays to implement load balancing across paths for different LUNs.

Change the Path Selection Policy

Generally, you do not have to change the default multipathing settings your host uses for a specific storage device. However, if you want to make any changes, you can use the Manage Paths dialog box to modify a path selection policy and specify the preferred path for the Fixed policy.

Procedure

1 Open the Manage Paths dialog box either from the Datastores or Devices view.

2 Select a path selection policy.

By default, VMware supports the following path selection policies. If you have a third-party PSP installed on your host, its policy also appears on the list.

- Fixed (VMware)
- Most Recently Used (VMware)
- Round Robin (VMware)

3 For the fixed policy, specify the preferred path by right-clicking the path you want to assign as the preferred path, and selecting **Preferred**.

4 Click **OK** to save your settings and exit the dialog box.

Disable Paths

You can temporarily disable paths for maintenance or other reasons. You can do so using the vSphere Client.

Procedure

1 Open the Manage Paths dialog box either from the Datastores or Devices view.

2 In the Paths panel, right-click the path to disable, and select **Disable**.

3 Click **OK** to save your settings and exit the dialog box.

You can also disable a path from the adapter's Paths view by right-clicking the path in the list and selecting **Disable**.

Managing Storage Paths and Multipathing Plug-Ins

Use the `esxcli` commands to manage the PSA multipathing plug-ins and storage paths assigned to them.

You can display all multipathing plug-ins available on your host. You can list any third-party MPPs, as well as your host's NMP and SATPs and review the paths they claim. You can also define new paths and specify which multipathing plug-in should claim the paths.

For more information about commands available to manage PSA, see the *Getting Started with vSphere Command-Line Interfaces*.

Multipathing Considerations

Specific considerations apply when you manage storage multipathing plug-ins and claim rules.

The following considerations help you with multipathing:

- If no SATP is assigned to the device by the claim rules, the default SATP for iSCSI or FC devices is VMW_SATP_DEFAULT_AA. The default PSP is VMW_PSP_FIXED.
- When the system searches the SATP rules to locate a SATP for a given device, it searches the driver rules first. If there is no match, the vendor/model rules are searched, and finally the transport rules are searched. If no match occurs, NMP selects a default SATP for the device.
- If VMW_SATP_ALUA is assigned to a specific storage device, but the device is not ALUA-aware, no claim rule match occurs for this device. The device is claimed by the default SATP based on the device's transport type.
- The default PSP for all devices claimed by VMW_SATP_ALUA is VMW_PSP_MRU. The VMW_PSP_MRU selects an active/optimized path as reported by the VMW_SATP_ALUA, or an active/unoptimized path if there is no active/optimized path. This path is used until a better path is available (MRU). For example, if the VMW_PSP_MRU is currently using an active/unoptimized path and an active/optimized path becomes available, the VMW_PSP_MRU will switch the current path to the active/optimized one.
- If you enable VMW_PSP_FIXED with VMW_SATP_ALUA, the host initially makes an arbitrary selection of the preferred path, regardless of whether the ALUA state is reported as optimized or unoptimized. As a result, VMware does not recommend to enable VMW_PSP_FIXED when VMW_SATP_ALUA is used for an ALUA-compliant storage array.

  The exception is when you assign the preferred path to be to one of the redundant storage processor (SP) nodes within an active-active storage array. The ALUA state is irrelevant.
- By default, the PSA claim rule 101 masks Dell array pseudo devices. Do not delete this rule, unless you want to unmask these devices.

List Multipathing Claim Rules for the Host

Use the `esxcli` command to list available multipathing claim rules.

Claim rules indicate which multipathing plug-in, the NMP or any third-party MPP, manages a given physical path. Each claim rule identifies a set of paths based on the following parameters:

- Vendor/model strings
- Transportation, such as SATA, IDE, Fibre Channel, and so on
- Adapter, target, or LUN location

- Device driver, for example, Mega-RAID

In the procedure, **--server=*server_name*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

- Run the **esxcli --server=*server_name* storage core claimrule list --claimrule-class=MP** command to list the multipathing claim rules.

Example: Sample Output of the esxcli storage core claimrule list Command

```
Rule Class  Rule   Class    Type       Plugin     Matches
MP          0      runtime  transport  NMP        transport=usb
MP          1      runtime  transport  NMP        transport=sata
MP          2      runtime  transport  NMP        transport=ide
MP          3      runtime  transport  NMP        transport=block
MP          4      runtime  transport  NMP        transport=unknown
MP          101    runtime  vendor     MASK_PATH  vendor=DELL model=Universal Xport
MP          101    file     vendor     MASK_PATH  vendor=DELL model=Universal Xport
MP          200    runtime  vendor     MPP_1      vendor=NewVend model=*
MP          200    file     vendor     MPP_1      vendor=NewVend model=*
MP          201    runtime  location   MPP_2      adapter=vmhba41 channel=* target=* lun=*
MP          201    file     location   MPP_2      adapter=vmhba41 channel=* target=* lun=*
MP          202    runtime  driver     MPP_3      driver=megaraid
MP          202    file     driver     MPP_3      driver=megaraid
MP          65535  runtime  vendor     NMP        vendor=* model=*
```

This example indicates the following:

- The NMP claims all paths connected to storage devices that use the USB, SATA, IDE, and Block SCSI transportation.
- You can use the MASK_PATH module to hide unused devices from your host. By default, the PSA claim rule 101 masks Dell array pseudo devices with a vendor string of DELL and a model string of Universal Xport.
- The MPP_1 module claims all paths connected to any model of the NewVend storage array.
- The MPP_3 module claims the paths to storage devices controlled by the Mega-RAID device driver.
- Any paths not described in the previous rules are claimed by NMP.
- The Rule Class column in the output describes the category of a claim rule. It can be MP (multipathing plug-in), Filter, or VAAI.
- The Class column shows which rules are defined and which are loaded. The `file` parameter in the Class column indicates that the rule is defined. The `runtime` parameter indicates that the rule has been loaded into your system. For a user-defined claim rule to be active, two lines with the same rule number should exist, one line for the rule with the `file` parameter and another line with `runtime`. Several low numbered rules, have only one line with the Class of `runtime`. These are system-defined claim rules that you cannot modify.

Display Multipathing Modules

Use the `esxcli` command to list all multipathing modules loaded into the system. Multipathing modules manage physical paths that connect your host with storage.

In the procedure, **`--server=`*`server_name`*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

- To list multipathing modules, run the following command:

  `esxcli --server=`*`server_name`*` storage core plugin list --plugin-class=MP`

This command typically shows the NMP and, if loaded, the MASK_PATH module. If any third-party MPPs have been loaded, they are listed as well.

Display SATPs for the Host

Use the `esxcli` command to list VMware NMP SATPs loaded into the system. Display information about the SATPs.

In the procedure, **`--server=`*`server_name`*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

- To list VMware SATPs, run the following command:

  `esxcli --server=`*`server_name`*` storage nmp satp list`

For each SATP, the output displays information that shows the type of storage array or system this SATP supports and the default PSP for any LUNs using this SATP. `Placeholder (plugin not loaded)` in the Description column indicates that the SATP is not loaded.

Display NMP Storage Devices

Use the `esxcli` command to list all storage devices controlled by the VMware NMP and display SATP and PSP information associated with each device.

In the procedure, **`--server=`*`server_name`*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

- To list all storage devices, run the following command:

  esxcli --server=*server_name* storage nmp device list

  Use the **--device | -d=*device_ID*** option to filter the output of this command to show a single device.

Add Multipathing Claim Rules

Use the `esxcli` commands to add a new multipathing PSA claim rule to the set of claim rules on the system. For the new claim rule to be active, you first define the rule and then load it into your system.

You add a new PSA claim rule when, for example, you load a new multipathing plug-in (MPP) and need to define which paths this module should claim. You may need to create a claim rule if you add new paths and want an existing MPP to claim them.

CAUTION When creating new claim rules, be careful to avoid a situation where different physical paths to the same LUN are claimed by different MPPs. Unless one of the MPPs is the MASK_PATH MPP, this configuration will cause performance problems.

In the procedure, **--server=*server_name*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

1 To define a new claim rule, run the following command:

  esxcli --server=*server_name* storage core claimrule add

  The command takes the following options:

Option	Description
-A|--adapter=<str>	Indicate the adapter of the paths to use in this operation.
-u|--autoassign	The system will auto assign a rule ID.
-C|--channel=<long>	Indicate the channel of the paths to use in this operation.
-c|--claimrule-class=<str>	Indicate the claim rule class to use in this operation. Valid values are: MP, Filter, VAAI.
-d|--device=<str>	Indicate the device Uid to use for this operation.
-D|--driver=<str>	Indicate the driver of the paths to use in this operation.
-f|--force	Force claim rules to ignore validity checks and install the rule anyway.
--if-unset=<str>	Execute this command if this advanced user variable is not set to 1.
-i|--iqn=<str>	Indicate the iSCSI Qualified Name for the target to use in this operation.
-L|--lun=<long>	Indicate the LUN of the paths to use in this operation.
-M|--model=<str>	Indicate the model of the paths to use in this operation.
-P|--plugin=<str>	Indicate which PSA plugin to use for this operation. (required)
-r|--rule=<long>	Indicate the rule ID to use for this operation.
-T|--target=<long>	Indicate the target of the paths to use in this operation.
-R|--transport=<str>	Indicate the transport of the paths to use in this operation. Valid values are: block, fc, iscsi, iscsivendor, ide, sas, sata, usb, parallel, unknown.

Option	Description
-t|--type=<str>	Indicate which type of matching is used for claim/unclaim or claimrule. Valid values are: vendor, location, driver, transport, device, target. (required)
-V|--vendor=<str>	Indicate the vendor of the paths to user in this operation.
--wwnn=<str>	Indicate the World-Wide Node Number for the target to use in this operation.
--wwpn=<str>	Indicate the World-Wide Port Number for the target to use in this operation.

2 To load the new claim rule into your system, run the following command:

esxcli --server=*server_name* storage core claimrule load

This command loads all newly created multipathing claim rules from your system's configuration file.

Example: Defining Multipathing Claim Rules

In the following example, you add and load rule # 500 to claim all paths with the NewMod model string and the NewVend vendor string for the NMP plug-in.

 # esxcli --server=*server_name* storage core claimrule add -r 500 -t vendor -V NewVend -M NewMod -P NMP

 # esxcli --server=*server_name* storage core claimrule load

After you run the **esxcli --server=*server_name* storage core claimrule list** command, you can see the new claim rule appearing on the list.

NOTE The two lines for the claim rule, one with the Class of `runtime` and another with the Class of `file`, indicate that the new claim rule has been loaded into the system and is active.

```
Rule Class  Rule   Class    Type       Plugin     Matches
MP          0      runtime  transport  NMP        transport=usb
MP          1      runtime  transport  NMP        transport=sata
MP          2      runtime  transport  NMP        transport=ide
MP          3      runtime  transport  NMP        transport=block
MP          4      runtime  transport  NMP        transport=unknown
MP          101    runtime  vendor     MASK_PATH  vendor=DELL model=Universal Xport
MP          101    file     vendor     MASK_PATH  vendor=DELL model=Universal Xport
MP          500    runtime  vendor     NMP        vendor=NewVend model=NewMod
MP          500    file     vendor     NMP        vendor=NewVend model=NewMod
```

Delete Multipathing Claim Rules

Use the `esxcli` commands to remove a multipathing PSA claim rule from the set of claim rules on the system.

In the procedure, **--server=*server_name*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

1 Delete a claim rule from the set of claim rules.

```
esxcli --server=server_name storage core claimrule remove
```

NOTE By default, the PSA claim rule 101 masks Dell array pseudo devices. Do not delete this rule, unless you want to unmask these devices.

The command takes the following options:

Option	Description
-c|--claimrule-class=<str>	Indicate the claim rule class to use in this operation (MP, Filter, VAAI).
-P|--plugin=<str>	Indicate the plugin to use for this operation.
-r|--rule=<long>	Indicate the rule ID to use for this operation.

This step removes the claim rule from the File class.

2 Remove the claim rule from the system.

```
esxcli --server=server_name storage core claimrule load
```

This step removes the claim rule from the Runtime class.

Mask Paths

You can prevent the host from accessing storage devices or LUNs or from using individual paths to a LUN. Use the `esxcli` commands to mask the paths. When you mask paths, you create claim rules that assign the MASK_PATH plug-in to the specified paths.

In the procedure, **--server=*server_name*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

1 Check what the next available rule ID is.

```
esxcli --server=server_name storage core claimrule list
```

The claim rules that you use to mask paths should have rule IDs in the range of 101 – 200. If this command shows that rule 101 and 102 already exist, you can specify 103 for the rule to add.

2 Assign the MASK_PATH plug-in to a path by creating a new claim rule for the plug-in.

```
esxcli --server=server_name storage core claimrule add -P MASK_PATH
```

3 Load the MASK_PATH claim rule into your system.

```
esxcli --server=server_name storage core claimrule load
```

4 Verify that the MASK_PATH claim rule was added correctly.

```
esxcli --server=server_name storage core claimrule list
```

5 If a claim rule for the masked path exists, remove the rule.

```
esxcli --server=server_name storage core claiming unclaim
```

6 Run the path claiming rules.

```
esxcli --server=server_name storage core claimrule run
```

After you assign the MASK_PATH plug-in to a path, the path state becomes irrelevant and is no longer maintained by the host. As a result, commands that display the masked path's information might show the path state as dead.

Example: Masking a LUN

In this example, you mask the LUN 20 on targets T1 and T2 accessed through storage adapters vmhba2 and vmhba3.

1 `#esxcli --server=server_name storage core claimrule list`

2
```
#esxcli --server=server_name storage core claimrule add -P MASK_PATH -r 109 -t location -A
vmhba2 -C 0 -T 1 -L 20
#esxcli --server=server_name storage core claimrule add -P MASK_PATH -r 110 -t location -A
vmhba3 -C 0 -T 1 -L 20
#esxcli --server=server_name storage core claimrule add -P MASK_PATH -r 111 -t location -A
vmhba2 -C 0 -T 2 -L 20
#esxcli --server=server_name storage core claimrule add -P MASK_PATH -r 112 -t location -A
vmhba3 -C 0 -T 2 -L 20
```

3 `#esxcli --server=server_name storage core claimrule load`

4 `#esxcli --server=server_name storage core claimrule list`

5
```
#esxcli --server=server_name storage core claiming unclaim -t location -A vmhba2
#esxcli --server=server_name storage core claiming unclaim -t location -A vmhba3
```

6 `#esxcli --server=server_name storage core claimrule run`

Unmask Paths

When you need the host to access the masked storage device, unmask the paths to the device.

In the procedure, **--server=*server_name*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

1 Delete the MASK_PATH claim rule.

```
esxcli --server=server_name storage core claimrule remove -r rule#
```

2 Verify that the claim rule was deleted correctly.

```
esxcli --server=server_name storage core claimrule list
```

3 Reload the path claiming rules from the configuration file into the VMkernel.

```
esxcli --server=server_name storage core claimrule load
```

4 Run the **esxcli --server=*server_name* storage core claiming unclaim** command for each path to the masked storage device.

For example:

```
esxcli --server=server_name storage core claiming unclaim -t location -A vmhba0 -C 0 -T 0 -L 149
```

5 Run the path claiming rules.

```
esxcli --server=server_name storage core claimrule run
```

Your host can now access the previously masked storage device.

Define NMP SATP Rules

The NMP SATP claim rules specify which SATP should manage a particular storage device. Usually you do not need to modify the NMP SATP rules. If you need to do so, use the `esxcli` commands to add a rule to the list of claim rules for the specified SATP.

You might need to create a SATP rule when you install a third-party SATP for a specific storage array.

In the procedure, **--server=*server_name*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

1 To add a claim rule for a specific SATP, run the **esxcli --server=*server_name* storage nmp satp rule add** command. The command takes the following options.

Option	Description
-b|--boot	This is a system default rule added at boot time. Do not modify esx.conf or add to host profile.
-c|--claim-option=*string*	Set the claim option string when adding a SATP claim rule.
-e|--description=*string*	Set the claim rule description when adding a SATP claim rule.
-d|--device=*string*	Set the device when adding SATP claim rules. Device rules are mutually exclusive with vendor/model and driver rules.
-D|--driver=*string*	Set the driver string when adding a SATP claim rule. Driver rules are mutually exclusive with vendor/model rules.
-f|--force	Force claim rules to ignore validity checks and install the rule anyway.
-h|--help	Show the help message.
-M|--model=*string*	Set the model string when adding SATP a claim rule. Vendor/Model rules are mutually exclusive with driver rules.
-o|--option=*string*	Set the option string when adding a SATP claim rule.
-P|--psp=*string*	Set the default PSP for the SATP claim rule.
-O|--psp-option=*string*	Set the PSP options for the SATP claim rule.
-s|--satp=*string*	The SATP for which a new rule will be added.
-R|--transport=*string*	Set the claim transport type string when adding a SATP claim rule.

Option	Description
-t|--type=*string*	Set the claim type when adding a SATP claim rule.
-V|--vendor=*string*	Set the vendor string when adding SATP claim rules. Vendor/Model rules are mutually exclusive with driver rules.

NOTE When searching the SATP rules to locate a SATP for a given device, the NMP searches the driver rules first. If there is no match, the vendor/model rules are searched, and finally the transport rules. If there is still no match, NMP selects a default SATP for the device.

2 Reboot your host.

Example: Defining an NMP SATP Rule

The following sample command assigns the VMW_SATP_INV plug-in to manage storage arrays with vendor string NewVend and model string NewMod.

```
# esxcli --server=server_name storage nmp satp rule add -V NewVend -M NewMod -s VMW_SATP_INV
```

If you run the **esxcli --server=*server_name* storage nmp satp list -s VMW_SATP_INV** command, you can see the new rule added to the list of VMW_SATP_INV rules.

18 Storage Hardware Acceleration

The hardware acceleration functionality enables the ESXi host to integrate with compliant storage arrays and offload specific virtual machine and storage management operations to storage hardware. With the storage hardware assistance, your host performs these operations faster and consumes less CPU, memory, and storage fabric bandwidth.

The hardware acceleration is supported by block storage devices, Fibre Channel and iSCSI, and NAS devices.

This chapter includes the following topics:

- "Hardware Acceleration Benefits," on page 173
- "Hardware Acceleration Requirements," on page 174
- "Hardware Acceleration Support Status," on page 174
- "Hardware Acceleration for Block Storage Devices," on page 174
- "Hardware Acceleration on NAS Devices," on page 179
- "Hardware Acceleration Considerations," on page 181

Hardware Acceleration Benefits

When the hardware acceleration functionality is supported, the host can get hardware assistance and perform several tasks faster and more efficiently.

The host can get assistance with the following activities:

- Migrating virtual machines with Storage vMotion
- Deploying virtual machines from templates
- Cloning virtual machines or templates
- VMFS clustered locking and metadata operations for virtual machine files
- Writes to thin provisioned and thick virtual disks
- Creating fault-tolerant virtual machines
- Creating and cloning thick disks on NFS datastores

Hardware Acceleration Requirements

The hardware acceleration functionality works only if you use an appropriate host and storage array combination.

Table 18-1. Hardware Acceleration Storage Requirements

ESXi	Block Storage Devices	NAS Devices
ESX/ESXi version 4.1	Support block storage plug-ins for array integration (VAAI)	Not supported
ESXi version 5.0	Support T10 SCSI standard or block storage plug-ins for array integration (VAAI)	Support NAS plug-ins for array integration

NOTE If your SAN or NAS storage fabric uses an intermediate appliance in front of a storage system that supports hardware acceleration, the intermediate appliance must also support hardware acceleration and be properly certified. The intermediate appliance might be a storage virtualization appliance, I/O acceleration appliance, encryption appliance, and so on.

Hardware Acceleration Support Status

For each storage device and datastore, the vSphere Client displays the hardware acceleration support status in the Hardware Acceleration column of the Devices view and the Datastores view.

The status values are Unknown, Supported, and Not Supported. The initial value is Unknown.

For block devices, the status changes to Supported after the host successfully performs the offload operation. If the offload operation fails, the status changes to Not Supported. The status remains Unknown if the device provides partial hardware acceleration support.

With NAS, the status becomes Supported when the storage can perform at least one hardware offload operation.

When storage devices do not support or provide partial support for the host operations, your host reverts to its native methods to perform unsupported operations.

Hardware Acceleration for Block Storage Devices

With hardware acceleration, your host can integrate with block storage devices, Fibre Channel or iSCSI, and use certain storage array operations.

ESXi hardware acceleration supports the following array operations:

- Full copy, also called clone blocks or copy offload. Enables the storage arrays to make full copies of data within the array without having the host read and write the data. This operation reduces the time and network load when cloning virtual machines, provisioning from a template, or migrating with vMotion.
- Block zeroing, also called write same. Enables storage arrays to zero out a large number of blocks to provide newly allocated storage, free of previously written data. This operation reduces the time and network load when creating virtual machines and formatting virtual disks.
- Hardware assisted locking, also called atomic test and set (ATS). Supports discrete virtual machine locking without use of SCSI reservations. This operation allows disk locking per sector, instead of the entire LUN as with SCSI reservations.

Check with your vendor for the hardware acceleration support. Certain storage arrays require that you activate the support on the storage side.

On your host, the hardware acceleration is enabled by default. If your storage does not support the hardware acceleration, you can disable it using the vSphere Client.

In addition to hardware acceleration support, ESXi includes support for array thin provisioning. For information, see "Array Thin Provisioning and VMFS Datastores," on page 186.

Disable Hardware Acceleration for Block Storage Devices

On your host, the hardware acceleration for block storage devices is enabled by default. You can use the vSphere Client advanced settings to disable the hardware acceleration operations.

As with any advanced settings, before you disable the hardware acceleration, consult with the VMware support team.

Procedure

1 In the vSphere Client inventory panel, select the host.

2 Click the **Configuration** tab, and click **Advanced Settings** under **Software.**

3 Change the value for any of the options to 0 (disabled):

- VMFS3.HardwareAcceleratedLocking
- DataMover.HardwareAcceleratedMove
- DataMover.HardwareAcceleratedInit

Managing Hardware Acceleration on Block Storage Devices

To integrate with the block storage arrays and to benefit from the array hardware operations, vSphere uses the ESXi extensions referred to as Storage APIs - Array Integration, formerly called VAAI.

In the vSphere 5.0 release, these extensions are implemented as the T10 SCSI based commands. As a result, with the devices that support the T10 SCSI standard, your ESXi host can communicate directly and does not require the VAAI plug-ins.

If the device does not support T10 SCSI or provides partial support, ESXi reverts to using the VAAI plug-ins, installed on your host, or uses a combination of the T10 SCSI commands and plug-ins. The VAAI plug-ins are vendor-specific and can be either VMware or partner developed. To manage the VAAI capable device, your host attaches the VAAI filter and vendor-specific VAAI plug-in to the device.

For information about whether your storage requires VAAI plug-ins or supports hardware acceleration through T10 SCSI commands, see the *vSphere Compatibility Guide* or check with your storage vendor.

You can use several `esxcli` commands to query storage devices for the hardware acceleration support information. For the devices that require the VAAI plug-ins, the claim rule commands are also available. For information about `esxcli` commands, see *Getting Started with vSphere Command-Line Interfaces*.

Display Hardware Acceleration Plug-Ins and Filter

To communicate with the devices that do not support the T10 SCSI standard, your host uses a combination of a single VAAI filter and a vendor-specific VAAI plug-in. Use the `esxcli` command to view the hardware acceleration filter and plug-ins currently loaded into your system.

In the procedure, **`--server=`*`server_name`*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

- Run the **esxcli --server=*server_name* storage core plugin list --plugin-class=*value*** command.

  For *value*, enter one of the following options:

  - Type `VAAI` to display plug-ins.

    The output of this command is similar to the following example:

    ```
    #esxcli --server=server_name storage core plugin list --plugin-class=VAAI
    Plugin name         Plugin class
    VMW_VAAIP_EQL       VAAI
    VMW_VAAIP_NETAPP    VAAI
    VMW_VAAIP_CX        VAAI
    ```

  - Type `Filter` to display the Filter.

    The output of this command is similar to the following example:

    ```
    esxcli --server=server_name storage core plugin list --plugin-class=Filter
    Plugin name  Plugin class
    VAAI_FILTER  Filter
    ```

Verify Hardware Acceleration Support Status

Use the `esxcli` command to verify the hardware acceleration support status of a particular storage device.

In the procedure, **--server=*server_name*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

- Run the **esxcli --server=*server_name* storage core device list -d=*device_ID*** command.

  The output shows the hardware acceleration, or VAAI, status that can be unknown, supported, or unsupported.

  ```
  # esxcli --server=server_name storage core device list -d naa.XXXXXXXXXXXX4c
  naa.XXXXXXXXXXXX4c
   Display Name: XXXX Fibre Channel Disk(naa.XXXXXXXXXXXX4c)
   Size: 20480
   Device Type: Direct-Access
   Multipath Plugin: NMP
   XXXXXXXXXXXXXXX
   Attached Filters: VAAI_FILTER
   VAAI Status: supported
   XXXXXXXXXXXXXXX
  ```

Verify Hardware Acceleration Support Details

Use the `esxcli` command to query the block storage device about the hardware acceleration support the device provides.

In the procedure, **--server=*server_name*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

- Run the **`esxcli --server=`*`server_name`*` storage core device vaai status get -d=`*`device_ID`* command.

  If the device is managed by a VAAI plug-in, the output shows the name of the plug-in attached to the device. The output also shows the support status for each T10 SCSI based primitive, if available. Output appears in the following example:

```
# esxcli --server=server_name storage core device vaai status get -d naa.XXXXXXXXXXXX4c
naa.XXXXXXXXXXX4c
 VAAI Plugin Name: VMW_VAAIP_SYMM
 ATS Status: supported
 Clone Status: supported
 Zero Status: supported
 Delete Status: unsupported
```

List Hardware Acceleration Claim Rules

Each block storage device managed by a VAAI plug-in needs two claim rules, one that specifies the hardware acceleration filter and another that specifies the hardware acceleration plug-in for the device. You can use the `esxcli` commands to list the hardware acceleration filter and plug-in claim rules.

Procedure

1 To list the filter claim rules, run the **`esxcli --server=`*`server_name`*` storage core claimrule list --claimrule-class=Filter`** command.

  In this example, the filter claim rules specify devices that should be claimed by the VAAI_FILTER filter.

```
# esxcli --server=server_name storage core claimrule list --claimrule-class=Filter
Rule Class   Rule   Class   Type   Plugin      Matches
Filter       65430  runtime vendor VAAI_FILTER vendor=EMC model=SYMMETRIX
Filter       65430  file    vendor VAAI_FILTER vendor=EMC model=SYMMETRIX
Filter       65431  runtime vendor VAAI_FILTER vendor=DGC model=*
Filter       65431  file    vendor VAAI_FILTER vendor=DGC model=*
```

2 To list the VAAI plug-in claim rules, run the **`esxcli --server=`*`server_name`*` storage core claimrule list --claimrule-class=VAAI`** command.

  In this example, the VAAI claim rules specify devices that should be claimed by a particular VAAI plug-in.

```
esxcli --server=server_name storage core claimrule list --claimrule-class=VAAI
Rule Class   Rule   Class   Type    Plugin         Matches
VAAI         65430  runtime vendor  VMW_VAAIP_SYMM vendor=EMC model=SYMMETRIX
VAAI         65430  file    vendor  VMW_VAAIP_SYMM vendor=EMC model=SYMMETRIX
VAAI         65431  runtime vendor  VMW_VAAIP_CX   vendor=DGC model=*
VAAI         65431  file    vendor  VMW_VAAIP_CX   vendor=DGC model=*
```

Add Hardware Acceleration Claim Rules

To configure hardware acceleration for a new array, you need to add two claim rules, one for the VAAI filter and another for the VAAI plug-in. For the new claim rules to be active, you first define the rules and then load them into your system.

This procedure is for those block storage devices that do not support T10 SCSI commands and instead use the VAAI plug-ins.

In the procedure, **--server=*server_name*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

1 Define a new claim rule for the VAAI filter by running the **esxcli --server=*server_name* storage core claimrule add --claimrule-class=Filter --plugin=VAAI_FILTER** command.

2 Define a new claim rule for the VAAI plug-in by running the **esxcli --server=*server_name* storage core claimrule add --claimrule-class=VAAI** command.

3 Load both claim rules by running the following commands:

```
esxcli --server=server_name storage core claimrule load --claimrule-class=Filter
esxcli --server=server_name storage core claimrule load --claimrule-class=VAAI
```

4 Run the VAAI filter claim rule by running the **esxcli --server=*server_name* storage core claimrule run --claimrule-class=Filter** command.

NOTE Only the Filter-class rules need to be run. When the VAAI filter claims a device, it automatically finds the proper VAAI plug-in to attach.

Example: Defining Hardware Acceleration Claim Rules

This example shows how to configure hardware acceleration for IBM arrays using the VMW_VAAIP_T10 plug-in. Use the following sequence of commands. For information about the options that the command takes, see "Add Multipathing Claim Rules," on page 167.

```
# esxcli --server=server_name storage core claimrule add --claimrule-class=Filter --plugin=VAAI_FILTER --type=vendor --vendor=IBM --autoassign
# esxcli --server=server_name storage core claimrule add --claimrule-class=VAAI --plugin=VMW_VAAIP_T10 --type=vendor --vendor=IBM --autoassign
# esxcli --server=server_name storage core claimrule load --claimrule-class=Filter
# esxcli --server=server_name storage core claimrule load --claimrule-class=VAAI
# esxcli --server=server_name storage core claimrule run --claimrule-class=Filter
```

Delete Hardware Acceleration Claim Rules

Use the `esxcli` command to delete existing hardware acceleration claim rules.

In the procedure, **--server=*server_name*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

- Run the following commands:

  ```
  esxcli --server=server_name storage core claimrule remove -r claimrule_ID --claimrule-class=Filter
  ```

  ```
  esxcli --server=server_name storage core claimrule remove -r claimrule_ID --claimrule-class=VAAI
  ```

Hardware Acceleration on NAS Devices

Hardware acceleration allows your host to integrate with NAS devices and use several hardware operations that NAS storage provides.

The following list shows the supported NAS operations:

- File clone. This operation is similar to the VMFS block cloning except that NAS devices clone entire files instead of file segments.
- Reserve space. Enables storage arrays to allocate space for a virtual disk file in thick format.

  Typically, when you create a virtual disk on an NFS datastore, the NAS server determines the allocation policy. The default allocation policy on most NAS servers is thin and does not guarantee backing storage to the file. However, the reserve space operation can instruct the NAS device to use vendor-specific mechanisms to reserve space for a virtual disk of nonzero logical size.
- Extended file statistics. Enables storage arrays to accurately report space utilization for virtual machines.

With NAS storage devices, the hardware acceleration integration is implemented through vendor-specific NAS plug-ins. These plug-ins are typically created by vendors and are distributed as VIB packages through a web page. No claim rules are required for the NAS plug-ins to function.

There are several tools available for installing and upgrading VIB packages. They include the `esxcli` commands and vSphere Update Manager. For more information, see the *vSphere Upgrade* and *Installing and Administering VMware vSphere Update Manager* documentation.

Install NAS Plug-In

Install vendor-distributed hardware acceleration NAS plug-ins on your host.

This topic provides an example for a VIB package installation using the `esxcli` command. For more details, see the *vSphere Upgrade* documentation.

In the procedure, **`--server=`*`server_name`*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

1 Place your host into the maintenance mode.

2 Set the host acceptance level:

  ```
  esxcli --server=server_name software acceptance set --level=value
  ```

  The command controls which VIB package is allowed on the host. The *value* can be one of the following:

  - VMwareCertified

- VMwareAccepted
- PartnerSupported
- CommunitySupported

3 Install the VIB package:

```
esxcli --server=server_name software vib install -v|--viburl=URL
```

The *URL* specifies the URL to the VIB package to install. http:, https:, ftp:, and file: are supported.

4 Verify that the plug-in is installed:

```
esxcli --server=server_name software vib list
```

5 Reboot your host for the installation to take effect.

Uninstall NAS Plug-Ins

To uninstall a NAS plug-in, remove the VIB package from your host.

This topic discusses how to uninstall a VIB package using the `esxcli` command. For more details, see the *vSphere Upgrade* documentation.

In the procedure, **`--server=`*`server_name`*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

1 Uninstall the plug-in:

```
esxcli --server=server_name software vib remove -n|--vibname=name
```

The *name* is the name of the VIB package to remove.

2 Verify that the plug-in is removed:

```
esxcli --server=server_name software vib list
```

3 Reboot your host for the change to take effect.

Update NAS Plug-Ins

Upgrade hardware acceleration NAS plug-ins on your host when a storage vendor releases a new plug-in version.

In the procedure, **`--server=`*`server_name`*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

This topic discusses how to update a VIB package using the `esxcli` command. For more details, see the *vSphere Upgrade* documentation.

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

1 Upgrade to a new plug-in version:

```
esxcli --server=server_name software vib update -v|--viburl=URL
```

The *URL* specifies the URL to the VIB package to update. http:, https:, ftp:, and file: are supported.

2 Verify that the correct version is installed:

```
esxcli --server=server_name software vib list
```

3 Reboot the host.

Verify Hardware Acceleration Status for NAS

In addition to the vSphere Client, you can use the `esxcli` command to verify the hardware acceleration status of the NAS device.

In the procedure, **`--server=`*`server_name`*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

- Run the **`esxcli --server=`*`server_name`*` storage nfs list`** command.

  The Hardware Acceleration column in the output shows the status.

Hardware Acceleration Considerations

When you use the hardware acceleration functionality, certain considerations apply.

Several reasons might cause a hardware-accelerated operation to fail.

For any primitive that the array does not implement, the array returns an error. The error triggers the ESXi host to attempt the operation using its native methods.

The VMFS data mover does not leverage hardware offloads and instead uses software data movement when one of the following occurs:

- The source and destination VMFS datastores have different block sizes.
- The source file type is RDM and the destination file type is non-RDM (regular file).
- The source VMDK type is eagerzeroedthick and the destination VMDK type is thin.
- The source or destination VMDK is in sparse or hosted format.
- The source virtual machine has a snapshot.
- The logical address and transfer length in the requested operation are not aligned to the minimum alignment required by the storage device. All datastores created with the vSphere Client are aligned automatically.
- The VMFS has multiple LUNs or extents, and they are on different arrays.

Hardware cloning between arrays, even within the same VMFS datastore, does not work.

19 Storage Thin Provisioning

With ESXi, you can use two models of thin provisioning, array-level and virtual disk-level.

Thin provisioning is a method that optimizes storage utilization by allocating storage space in a flexible on-demand manner. Thin provisioning contrasts with the traditional model, called thick provisioning. With thick provisioning, large amount of storage space is provided in advance in anticipation of future storage needs. However, the space might remain unused causing underutilization of storage capacity.

The VMware thin provisioning features help you eliminate storage underutilization problems at the datastore and storage array level.

This chapter includes the following topics:

- "Storage Over-Subscription," on page 183
- "Virtual Disk Thin Provisioning," on page 183
- "Array Thin Provisioning and VMFS Datastores," on page 186

Storage Over-Subscription

Thin provisioning allows you to report more virtual storage space than there is real physical capacity. This discrepancy can lead to storage over-subscription, also called over-provisioning.

When you use thin provisioning, you should monitor actual storage usage to avoid conditions when you run out of physical storage space.

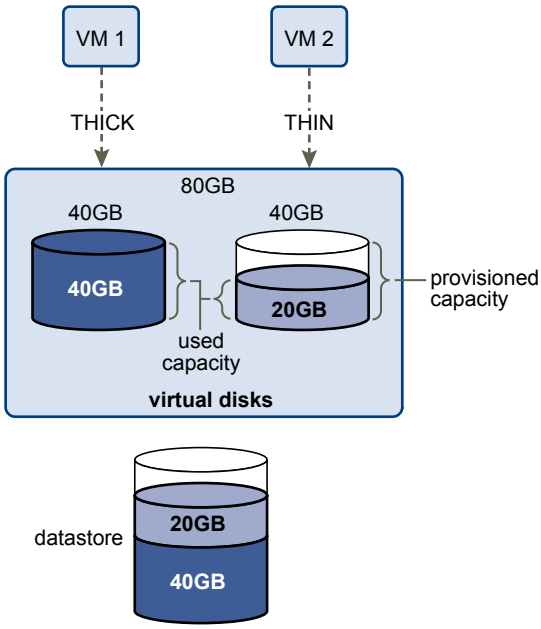
Virtual Disk Thin Provisioning

When you create a virtual machine, a certain amount of storage space on a datastore is provisioned to virtual disk files.

By default, ESXi offers a traditional storage provisioning method for virtual machines. With this method, you first estimate how much storage the virtual machine will need for its entire life cycle. You then provision a fixed amount of storage space to its virtual disk in advance, for example, 40GB, and have the entire provisioned space committed to the virtual disk. A virtual disk that immediately occupies the entire provisioned space is a thick disk.

ESXi supports thin provisioning for virtual disks. With the disk-level thin provisioning feature, you can create virtual disks in a thin format. For a thin virtual disk, ESXi provisions the entire space required for the disk's current and future activities, for example 40GB. However, the thin disk commits only as much storage space as the disk needs for its initial operations. In this example, the thin-provisioned disk occupies only 20GB of storage. As the disk requires more space, it can grow into its entire 40GB provisioned space.

Figure 19-1. Thick and thin virtual disks



Create Thin Provisioned Virtual Disks

When you need to save storage space, you can create a virtual disk in thin provisioned format. The thin provisioned virtual disk starts small and grows as more disk space is required.

This procedure assumes that you are creating a typical or custom virtual machine using the New Virtual Machine wizard.

Prerequisites

You can create thin disks only on the datastores that support disk-level thin provisioning.

Procedure

- In the Create a Disk dialog box, select **Thin Provision**.

A virtual disk in thin format is created.

What to do next

If you created a virtual disk in the thin format, you can later inflate it to its full size.

About Virtual Disk Provisioning Policies

When you perform certain virtual machine management operations, such as creating a virtual disk, cloning a virtual machine to a template, or migrating a virtual machine, you can specify a provisioning policy for the virtual disk file.

NFS datastores with Hardware Acceleration and VMFS datastores support the following disk provisioning policies. On NFS datastores that do not support Hardware Acceleration, only thin format is available.

You can use Storage vMotion to transform virtual disks from one format to another.

Thick Provision Lazy Zeroed — Creates a virtual disk in a default thick format. Space required for the virtual disk is allocated when the virtual disk is created. Data remaining on the physical device is not erased during creation, but is zeroed out on demand at a later time on first write from the virtual machine.

Using the default flat virtual disk format does not zero out or eliminate the possibility of recovering deleted files or restoring old data that might be present on this allocated space. You cannot convert a flat disk to a thin disk.

Thick Provision Eager Zeroed A type of thick virtual disk that supports clustering features such as Fault Tolerance. Space required for the virtual disk is allocated at creation time. In contrast to the flat format, the data remaining on the physical device is zeroed out when the virtual disk is created. It might take much longer to create disks in this format than to create other types of disks.

Thin Provision Use this format to save storage space. For the thin disk, you provision as much datastore space as the disk would require based on the value that you enter for the disk size. However, the thin disk starts small and at first, uses only as much datastore space as the disk needs for its initial operations.

NOTE If a virtual disk supports clustering solutions such as Fault Tolerance, do not make the disk thin.

If the thin disk needs more space later, it can grow to its maximum capacity and occupy the entire datastore space provisioned to it. Also, you can manually convert the thin disk into a thick disk.

View Virtual Machine Storage Resources

You can view how datastore storage space is allocated for your virtual machines.

Procedure

1 Select the virtual machine in the inventory.

2 Click the **Summary** tab.

3 Review the space allocation information in the Resources section.

- Provisioned Storage – Shows datastore space guaranteed to the virtual machine. The entire space might not be used by the virtual machine if it has disks in thin provisioned format. Other virtual machines can occupy any unused space.
- Not-shared Storage – Shows datastore space occupied by the virtual machine and not shared with any other virtual machines.
- Used Storage – Shows datastore space actually occupied by virtual machine files, including configuration and log files, snapshots, virtual disks, and so on. When the virtual machine is running, the used storage space also includes swap files.

Determine the Disk Format of a Virtual Machine

You can determine whether your virtual disk is in thick or thin format.

Procedure

1 Select the virtual machine in the inventory.

2 Click **Edit Settings** to display the Virtual Machine Properties dialog box.

3 Click the **Hardware** tab and select the appropriate hard disk in the Hardware list.

The Disk Provisioning section on the right shows the type of your virtual disk.

4 Click **OK**.

What to do next

If your virtual disk is in the thin format, you can inflate it to its full size.

Inflate Thin Virtual Disks

If you created a virtual disk in the thin provision format, you can inflate it to its full size.

This procedure converts a thin disk to a virtual disk in thick provision format.

Procedure

1 Select the virtual machine in the inventory.

2 Click the **Summary** tab and, under Resources, double-click the datastore for the virtual machine to open the Datastore Browser dialog box.

3 Click the virtual machine folder to find the virtual disk file you want to convert. The file has the `.vmdk` extension.

4 Right-click the virtual disk file and select **Inflate**.

The inflated virtual disk occupies the entire datastore space originally provisioned to it.

Handling Datastore Over-Subscription

Because the provisioned space for thin disks can be greater than the committed space, a datastore over-subscription can occur, which results in the total provisioned space for the virtual machine disks on the datastore being greater than the actual capacity.

Over-subscription can be possible because usually not all virtual machines with thin disks need the entire provisioned datastore space simultaneously. However, if you want to avoid over-subscribing the datastore, you can set up an alarm that notifies you when the provisioned space reaches a certain threshold.

For information on setting alarms, see the *vCenter Server and Host Management* documentation.

If your virtual machines require more space, the datastore space is allocated on a first come first served basis. When the datastore runs out of space, you can add more physical storage and increase the datastore.

See Increase VMFS Datastores.

Array Thin Provisioning and VMFS Datastores

You can use thin provisioned storage arrays with ESXi.

Traditional LUNs that arrays present to the ESXi host, are thick-provisioned. The entire physical space needed to back each LUN is allocated in advance.

ESXi also supports thin-provisioned LUNs. When a LUN is thin-provisioned, the storage array reports the LUN's logical size, which might be larger than the real physical capacity backing that LUN.

A VMFS datastore that you deploy on the thin-provisioned LUN can detect only the logical size of the LUN. For example, if the array reports 2TB of storage while in reality the array provides only 1TB, the datastore considers 2TB to be the LUN's size. As the datastore grows, it cannot determine whether the actual amount of physical space is still sufficient for its needs.

However, when you use the Storage APIs - Array Integration, the host can integrate with physical storage and become aware of underlying thin-provisioned LUNs and their space usage.

Using thin provision integration, your host can perform these tasks:

- Monitor the use of space on thin-provisioned LUNs to avoid running out of physical space. As your datastore grows or if you use Storage vMotion to migrate virtual machines to a thin-provisioned LUN, the host communicates with the LUN and warns you about breaches in physical space and about out-of-space conditions.
- Inform the array about the datastore space that is freed when files are deleted or removed from the datastore by Storage vMotion. The array can then reclaim the freed blocks of space.

NOTE ESXi does not support enabling and disabling of thin provisioning on a storage device.

Requirements

To use the thin provision reporting feature, your host and storage array must meet the following requirements:

- ESXi version 5.0 or later.
- Storage array has appropriate firmware that supports T10-based Storage APIs - Array Integration (Thin Provisioning). For information, contact your storage provider and check the HCL.

Space Usage Monitoring

The thin provision integration functionality helps you to monitor the space usage on thin-provisioned LUNs and to avoid running out of space.

The following sample flow demonstrates how the ESXi host and the storage array interact to generate breach of space and out-of-space warnings for a datastore with underlying thin-provisioned LUN. The same mechanism applies when you use Storage vMotion to migrate virtual machines to the thin-provisioned LUN.

1. Using storage-specific tools, your storage administrator provisions a thin LUN and sets a soft threshold limit that, when reached, triggers an alert. This step is vendor-specific.
2. Using the vSphere Client, you create a VMFS datastore on the thin-provisioned LUN. The datastore spans the entire logical size that the LUN reports.
3. As the space used by the datastore increases and reaches the specified soft threshold, the following actions take place:
   a. The storage array reports the breach to your host.
   b. Your host triggers a warning alarm for the datastore.

      You can contact the storage administrator to request more physical space or use Storage vMotion to evacuate your virtual machines before the LUN runs out of capacity.
4. If no space is left to allocate to the thin-provisioned LUN, the following actions take place:
   a. The storage array reports out-of-space condition to your host.
   b. The host pauses virtual machines and generates an out-of-space alarm.

      You can resolve the permanent out-of-space condition by requesting more physical space from the storage administrator.

Identify Thin-Provisioned Storage Devices

Use the `esxcli` command to verify whether a particular storage device is thin-provisioned.

In the procedure, **`--server=`*`server_name`*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

- Run the **`esxcli --server=`*`server_name`*` storage core device list -d=`*`device_ID`* command.

The following thin provisioning status indicates that the storage device is thin-provisioned.

```
# esxcli --server=server_name storage core device list -d naa.XXXXXXXXXXX4c
naa.XXXXXXXXXXX4c
 Display Name: XXXX Fibre Channel Disk(naa.XXXXXXXXXXX4c)
 Size: 20480
 Device Type: Direct-Access
 Multipath Plugin: NMP
 ---------------------
 Thin Provisioning Status: yes
 Attached Filters: VAAI_FILTER
 VAAI Status: supported
 ---------------------
```

An unknown status indicates that a storage device is thick.

NOTE Some storage systems present all devices as thin-provisioned no matter whether the devices are thin or thick. Their thin provisioning status is always `yes`. For details, check with your storage vendor.

Disable Space Reclamation

Use the `esxcli` command to disable space reclamation on a thin-provisioned LUN.

When you delete virtual machine files from a VMFS datastore, or migrate them through Storage vMotion, the datastore frees blocks of space and informs the storage array, so that the blocks can be reclaimed. If you do not need thin provisioned LUNs to reclaim the freed space, you can disable space reclamation on the host.

NOTE As with any advanced settings, before disabling the space reclamation, consult with the VMware support team.

In the procedure, **`--server=`*`server_name`*** specifies the target server. The specified target server prompts you for a user name and password. Other connection options, such as a configuration file or session file, are supported. For a list of connection options, see *Getting Started with vSphere Command-Line Interfaces*.

Prerequisites

Install vCLI or deploy the vSphere Management Assistant (vMA) virtual machine. See *Getting Started with vSphere Command-Line Interfaces*. For troubleshooting , run `esxcli` commands in the ESXi Shell.

Procedure

- Run the following command:
  `esxcli --server=`*`server_name`*` system settings advanced set --int-value 0 --option /VMFS3/EnableBlockDelete`.

Reclaim Accumulated Storage Space

If you used a thin-provisioned storage device with a legacy ESX/ESXi host (version prior to 5.0), you can retroactively reclaim accumulated free space for the device.

Prerequisites

- Upgrade your host to ESXi 5.0.
- Upgrade a datastore deployed on the thin-provisioned device to VMFS5.

Procedure

1 Change to the datastore directory by using the **`cd /vmfs/volumes/`*`datastore_name`***.

2 Reclaim a specified percentage of free capacity on the VMFS5 datastore for the thin-provisioned device by running **`vmkfstools –y`** ***`%`***.

% is a number between 1 and 99. The number represents a percentage of VMFS free capacity to be reclaimed. The recommended value is 60.

Using Storage Vendor Providers 20

When using vendor provider components, the vCenter Server can integrate with external storage, both block storage and NFS, so that you can gain a better insight into resources and obtain comprehensive and meaningful storage data.

The vendor provider is a software plug-in developed by a third party through the Storage APIs - Storage Awareness. The vendor provider component is typically installed on the storage array side and acts as a server in the vSphere environment. The vCenter Server uses vendor providers to retrieve information about storage topology, capabilities, and status.

For information about whether your storage supports the vendor provider plug-ins, contact your storage vendor.

If your storage supports vendor providers, use the **Storage Providers** menu option in the vSphere Client to register and manage each vendor provider component.

This chapter includes the following topics:

- "Vendor Providers and Storage Data Representation," on page 191
- "Vendor Provider Requirements and Considerations," on page 192
- "Storage Status Reporting," on page 192
- "Register Vendor Providers," on page 193
- "View Vendor Provider Information," on page 193
- "Unregister Vendor Providers," on page 194
- "Update Vendor Providers," on page 194

Vendor Providers and Storage Data Representation

The vCenter Server communicates with the vendor provider to obtain information that the vendor provider collects from available storage devices. The vCenter Server can then display the storage data in the vSphere Client.

Information that the vendor provider supplies can be divided into the following categories:

- Storage topology. Information about physical storage elements appears on the Storage Views tab. It includes such data as storage arrays, array IDs, and so on.

  This type of information can be helpful when you need to track virtual machine-to-storage relationships and configuration, or to identify any changes in physical storage configuration.

  For more information, see the *vSphere Monitoring and Performance* documentation.

- Storage capabilities. The vendor provider collects and communicates information about physical capabilities and services that underlying storage offers.

  This information can be useful when, for example, you need to properly aggregate storage into tiers, or select the right storage, in terms of space and performance, for a particular virtual machine.

  The capabilities appear on the list of system-defined storage capabilities. For details, see "Understanding Storage Capabilities," on page 195.

- Storage status. This category includes reporting about status of various storage entities. It also includes alarms and events for notifying about configuration changes.

  This type of information can help you troubleshoot storage connectivity and performance problems. It can also help you to correlate array-generated events and alarms to corresponding performance and load changes on the array.

Vendor Provider Requirements and Considerations

When you use the vendor provider functionality, certain requirements and considerations apply.

The vendor provider functionality is implemented as an extension to the VMware vCenter Storage Monitoring Service (SMS). Because the SMS is a part of the vCenter Server, the vendor provider functionality does not need special installation or enablement on the vCenter Server side.

To use vendor providers, follow these requirements:

- vCenter Server version 5.0 or later.
- ESX/ESXi hosts version 4.0 or later.
- Storage arrays that support Storage APIs - Storage Awareness plug-ins. The vendor provider component must be installed on the storage side. See the *vSphere Compatibility Guide* or check with your storage vendor.

NOTE Fibre Channel over Ethernet (FCoE) does not support vendor providers.

The following considerations exist when you use the vendor providers:

- Both block storage and file system storage devices can use vendor providers.
- Vendor providers can run anywhere, except the vCenter Server.
- Multiple vCenter Servers can simultaneously connect to a single instance of a vendor provider.
- A single vCenter Server can simultaneously connect to multiple different vendor providers. It is possible to have a different vendor provider for each type of physical storage device available to your host.

Storage Status Reporting

If you use vendor providers, the vCenter Server can collect status characteristics for physical storage devices and display this information in the vSphere Client.

The status information includes events and alarms.

- Events indicate important changes in the storage configuration. Such changes might include creation and deletion of a LUN, or a LUN becoming inaccessible due to LUN masking.

  For a standalone host, the vSphere Client displays storage events in the Events tab. For managed hosts, information is displayed in the Tasks & Events tab.

- Alarms indicate a change in storage system availability. For example, when you use profile-based storage management, you can specify virtual machine storage requirements. When changes to underlying storage occur that might violate the storage requirements of the virtual machine, an alarm gets triggered.

For more information about events and alarms, see the *vSphere Monitoring and Performance* documentation.

Thin-provisioned LUNs have special reporting requirements. For information about space monitoring on thin-provisioned LUNs, see "Array Thin Provisioning and VMFS Datastores," on page 186.

Register Vendor Providers

To establish a connection between the vCenter Server and a vendor provider, you must register the vendor provider.

Prerequisites

Verify that the vendor provider component is installed on the storage side and obtain its credentials from your storage administrator.

Use the vSphere Client to connect to the vCenter Server system.

Procedure

1 Select **View > Administration > Storage Providers**.

2 Click **Add**.

3 In the **Add Vendor Provider** dialog box, type connection information for the vendor provider, including the name, URL, and credentials.

4 (Optional) To direct the vCenter Server to the vendor provider certificate, select the **Use Vendor Provider Certificate** option and specify the certificate's location.

If you do not select this option, the vSphere Client displays a thumbprint of the certificate. You can check the thumbprint and approve it.

5 Click **OK** to complete the registration.

The vCenter Server has registered the vendor provider and established a secure SSL connection with it.

SecuringCommunication with Vendor Providers

To communicate with a vendor provider, the vCenter Server uses a secure SSL connection. The SSL authentication mechanism requires that both parties, the vCenter Server and the vendor provider, exchange SSL certificates and add them to their truststores.

The vCenter Server can add the vendor provider certificate to its truststore as part of the vendor provider installation. If the certificate is not added during the installation, use one of the following methods to add it when registering the vendor provider:

- Direct the vCenter Server to the vendor provider certificate. In the **Add Vendor Provider** dialog box, select the **Use Vendor Provider Certificate** option and specify the certificate's location.
- Use a thumbprint of the vendor provider certificate. If you do not direct the vCenter Server to use the provider certificate, the vSphere Client displays the certificate thumbprint. You can check the thumbprint and approve it. The vCenter Server adds the certificate to the truststore and proceeds with the connection.

The vendor provider adds the vCenter Server certificate to its truststore when the vCenter Server first connects to the provider.

View Vendor Provider Information

After you register a vendor provider component with the vCenter Server, the vendor provider appears on the vendor providers list in the vSphere Client.

View general vendor provider information and details for each vendor component.

Procedure

1 Select **View > Administration > Storage Providers**.

2 In the Vendor Providers list, view the vendor provider components registered with the vCenter Server.

The list shows general vendor information including the name, URL, and the time of the last view refresh.

3 To display additional details, select a specific vendor provider from the list.

The details include storage array vendors and array models that the vendor provider supports.

NOTE A single vendor provider can support storage arrays from multiple different vendors.

Unregister Vendor Providers

Unregister vendor providers that you do not need.

Procedure

1 Select **View > Administration > Storage Providers**.

2 From the list of vendor providers, select the one you want to unregister and click **Remove**.

The vCenter Server terminates the connection and removes the vendor provider from its configuration.

Update Vendor Providers

The vCenter Server periodically updates storage data in its database. The updates are partial and reflect only those storage changes that storage providers communicate to the vCenter Server. When needed, you can perform a full database synchronisation for the selected storage provider.

Procedure

1 Select **View > Administration > Storage Providers**.

2 From the list, select the vendor provider that you want to synchronise with and click **Sync**.

The vSphere Client updates the storage data for the provider.

Virtual Machine Storage Profiles 21

With profile-driven storage, you use storage capabilities and virtual machine storage profiles to ensure that virtual machines use storage that guarantees a certain level of capacity, performance, availability, redundancy, and so on.

NOTE Profile-driven storage does not support RDMs.

To manage storage placement by using virtual machine storage profiles, you must perform the following tasks:

1 Verify that system-defined storage capabilities appear in the Manage Storage Capabilities dialog box, if your storage system supports the Storage APIs - Storage Awareness.

   For more information about Storage APIs - Storage Awareness, see Chapter 20, "Using Storage Vendor Providers," on page 191.

2 Create user-defined storage capabilities.

3 Associate user-defined storage capabilities with datastores.

4 Enable virtual machine storage profiles for a host or cluster.

5 Create virtual machine storage profiles by defining the storage capabilities that an application running on a virtual machine requires.

6 Associate a virtual machine storage profile with the virtual machine files or virtual disks.

7 Verify that virtual machines and virtual disks use datastores that are compliant with their associated virtual machine storage profile.

This chapter includes the following topics:

- "Understanding Storage Capabilities," on page 195
- "Understanding Virtual Machine Storage Profiles," on page 198

Understanding Storage Capabilities

A storage capability outlines the quality of service that a storage system can deliver. It is a guarantee that the storage system can provide a specific set of characteristics for capacity, performance, availability, redundancy, and so on.

If a storage system uses Storage APIs - Storage Awareness, it informs vCenter Server that it can guarantee a specific set of storage features by presenting them as a storage capability. vCenter Server recognizes the capability and adds it to the list of storage capabilities in the Manage Storage Capabilities dialog box. Such storage capabilities are system-defined. vCenter Server assigns the system-defined storage capability to each datastore that you create from that storage system.

You can create user-defined storage capabilities and associate them with datastores. You should associate the same user-defined capability with datastores that guarantee the same level of storage capabilities. You can associate a user-defined capability with a datastore that already has a system-defined capability. A datastore can have only one system-defined and only one user-defined capability at a time.

For more information about Storage APIs - Storage Awareness, see Chapter 20, "Using Storage Vendor Providers," on page 191.

You define storage requirements for virtual machines and virtual disks by adding storage capabilities to virtual machine storage profiles.

View Existing Storage Capabilities

Before you add your storage capabilities, you can view system-defined storage capabilities that your storage system defines.

To view system-defined storage capabilities, your storage system must use Storage APIs - Storage Awareness.

Procedure

1 In the vSphere Client, select **View > Management > VM Storage Profiles**.

2 In the VM Storage Profiles view of the vSphere Client, click **Manage Storage Capabilities**.

The Manage Storage Capabilities dialog box appears.

3 View the names and descriptions of the storage capabilities in the Name and Description column.

4 View the System and User-defined types of the existing storage capabilities.

a Click the **Type** column to sort the storage capabilities by type.

b View the storage capabilities whose type is System.

c View the storage capabilities whose type is User-defined.

What to do next

Modify the list of existing user-defined storage capabilities by using the **Add**, **Remove**, or **Edit** buttons.

Add a User-Defined Storage Capability

You can create a storage capability and assign it to a datastore to indicate the capabilities that this datastore has.

Procedure

1 In the VM Storage Profiles view of the vSphere Client, click **Manage Storage Capabilities**.

The Manage Storage Capabilities dialog box appears.

2 Click **Add**.

3 Provide a name and a description for the storage capability.

Storage Capability Property	Example
Name	Fault tolerance.
Description	Storage that has a capacity over 2TB and is fault-tolerant.

4 Click **OK**.

The storage capability appears in the list and is specified as User-defined in the Type column.

What to do next

Assign the user-defined storage capabilities to datastores that have that capability.

Edit the Description of a User-Defined Storage Capability

You can edit a user-defined storage capability to make its description illustrate the quality of service that the storage capability guarantees.

You cannot edit system-defined storage capabilities.

Procedure

1 In the VM Storage Profiles view of the vSphere Client, click **Manage Storage Capabilities**.

The Manage Storage Capabilities dialog box appears.

2 In the Manage Storage Capabilities dialog box, select a user-defined storage capability and click **Edit**.

3 Provide a more detailed description of the storage capability and click **OK**.

You cannot change the name of a user-defined storage capability.

The new description appears in the Description column of the Manage Storage Capabilities dialog box.

Associate a User-Defined Storage Capability with a Datastore

After you create user-defined storage capabilities, you can associate them with datastores.

Whether a datastore has a system-defined storage capability or not, you can assign a user-defined storage capability to it. A datastore can have only one user-defined and only one system-defined storage capability at a time.

You cannot assign a user-defined storage capability to a datastore cluster. However, a datastore cluster inherits a system-defined or user-defined storage capabilities when all its datastores have the same system-defined or user-defined storage capability.

Prerequisites

Add a user-defined storage capability to the list of storage capabilities.

Procedure

1 In the vSphere Client, select **View > Inventory > Datastores and Datastore Clusters**.

2 Right-click a datastore from the inventory and select **Assign User-Defined Storage Capability**.

3 Select a storage capability from the list of storage capabilities and click **OK**.

Option	Description
Create and assign a new user-defined storage capability	a Click **New**. b Type a name and a description. c Click **OK**.
Select an existing storage capability	Select a user-defined storage capability from the **Name** drop-down menu and click **OK**.

The user-defined storage capability appears in the Storage Capabilities pane of the **Summary** tab of the datastore or its datastore cluster.

Associate a User-Defined Storage Capability with Multiple Datastores

You can select multiple datastores and associate them with a user-defined storage capability.

Procedure

1 Open a list of datastores in the vSphere Client.

Option	Action
View the datastores in a datacenter	a the Datastores and Datastore Clusters view, select a datacenter. b Select the **Datastores and Datastore Cluster** tab.
View the datastores mounted on a host	a In the Hosts and Clusters view, select a host. b Select the **Configuration** tab, and click **Storage**.

2 Press Ctrl and select several datastores.

3 Right-click a selected datastore and select **Assign User-Defined Storage Capability**.

4 Select a storage capability from the list of storage capabilities and click **OK**.

Option	Description
Create and assign a new user-defined storage capability	a Click **New**. b Type a name and a description. c Click **OK**.
Select an existing storage capability	Select a user-defined storage capability from the **Name** drop-down menu and click **OK**.

The storage capability is now associated with the selected datastores.

Remove a Storage Capability

You can remove an existing user-defined storage capability if you do not use it.

You can only remove user-defined storage capabilities. You cannot remove a system-defined storage capability.

CAUTION If you remove a storage capability that is part of a virtual machine storage profile, you might break the virtual machine storage profile compliance for the virtual machines and virtual disks that use it.

Procedure

1 In the VM Storage Profiles view of the vSphere Client, click **Manage Storage Capabilities**.

The Manage Storage Capabilities dialog box appears.

2 Select an unused storage capability and click **Remove**.

3 Click **Yes** in the confirmation dialog box.

The storage capability is removed from the list.

Understanding Virtual Machine Storage Profiles

Virtual machine storage profiles list the storage capabilities that virtual machine home files and virtual disks require to run the applications within the virtual machine.

You can create a list of virtual machine storage profiles to define different levels of storage requirements.

The virtual machine home files (`.vmx`, `.vmsd`, `.nvram`, `.log`, and so on) and the virtual disks (`.vmdk`) can have separate virtual machine storage profiles as shown in the following table.

Table 21-1. Example Virtual Machine Storage Profiles for a Virtual Machine

Example Virtual Machine Files	Example for a VM Storage Profile	Example for a Datastore Compliant with the VM Storage Profile
windows_2008r2_test.vmx	Storage Profile 2	datastore02
windows_2008r2_test.vmxf		
windows_2008r2_test.log		
windows_2008r2_test.nvram		
windows_2008r2_test.vmem		
windows_2008r2_test.vmsd		
windows_2008r2_test.vmdk	Storage Profile 3	datastore05
windows_2008r2_test_1.vmdk	Storage Profile 5	datastore10

When you create, clone, or migrate a virtual machine, you can select to associate it with a virtual machine storage profile. When you select a virtual machine storage profile, vSphere Client shows you the datastores that are compatible with the capabilities of the profile. You can then select a datastore or a datastore cluster.

If you select a datastore that does not match the virtual machine storage profile, the vSphere Client shows that the virtual machine is using non-compliant storage.

Enable Virtual Machine Storage Profiles on a Host or a Cluster

Before you can use virtual machine storage profiles, you must enable them on a host or a cluster.

Prerequisites

Verify that the host or all the hosts in the cluster for which you want to enable virtual machine storage profiles are licensed with a vSphere Enterprise Plus license key.

Procedure

1 In the VM Storage Profiles view of the vSphere Client, click **Enable VM Storage Profiles**.

The Enable VM Storage Profiles window appears. The window shows all available clusters and hosts, their licensing status, and whether virtual machine storage profiles are enabled or disabled for the host or the cluster.

2 To enable virtual machine storage profiles, select a host or a cluster whose status is Disabled or Unknown and click **Enable**.

The status of the host or cluster changes to Enabled in the VM Storage Profile Status column.

What to do next

You can use virtual machine storage profiles for the virtual machines that run on the enabled host or cluster.

Create a Virtual Machine Storage Profile

You can create a virtual machine storage profile to define storage requirements for a virtual machine and its virtual disks by listing storage capabilities in the virtual machine storage profile.

Prerequisites

Verify that you have at least one storage capability by clicking the **Manage Storage Capabilities** button and viewing the list of storage capabilities.

Procedure

1 In the VM Storage Profiles view of the vSphere Client, click **Create VM Storage Profile**.

The Create New VM Storage Profile wizard appears.

2 On the Profile Properties page, type a name and a description for the virtual machine storage profile, and click **Next**.

3 On the Select Storage Capabilities page, define storage requirements for the virtual machine storage profile by selecting one or more storage capabilities from the list, and click **Next**.

A datastore that has any of the selected capabilities will be compliant with the virtual machine storage profile.

4 On the Ready to Complete page, verify the virtual machine storage profile settings, and click **Finish**.

The new virtual machine storage profile appears one level under the `VM Storage Profiles` folder in the inventory.

What to do next

Apply the virtual machine storage profile to a virtual machine and its virtual disks.

Edit a Virtual Machine Storage Profile

You can change the storage requirements for virtual machines and virtual disks by modifying the list of storage capabilities that are associated with a virtual machine storage profile.

Procedure

1 In the VM Storage Profiles view of the vSphere Client, select an existing virtual machine storage profile from the inventory and click **Edit VM Storage Profile**.

The Edit VM Storage Profile dialog box opens.

2 Modify the virtual machine storage profile and click **OK**.

- Edit the name.
- Edit the description.
- Select new storage capabilities that you want to include in the virtual machine storage profile.
- Deselect storage capabilities that you want to exclude from the virtual machine storage profile.

IMPORTANT Excluding a storage capability might break the virtual machine storage profile compliance for a virtual machine, if the virtual machine or its disks use datastores with that capability.

The virtual machine storage profile is now changed.

Delete a Virtual Machine Storage Profile

You can delete a virtual machine storage profile if you are not using it for any virtual machine or virtual disk.

Procedure

1 In the VM Storage Profiles view of the vSphere Client, select a virtual machine storage profile that you do not use, and click **Delete VM Storage Profile**.

2 Click **Yes** in the confirmation dialog box.

The virtual machine storage profile is removed from the inventory.

Associate a Virtual Machine Storage Profile with a Virtual Machine and Its Virtual Disks

You can associate a virtual machine storage profile with a virtual machine to define the storage capabilities that are required by the applications running on the virtual machine.

You can associate a virtual machine storage profile with a powered-off and powered-on virtual machine.

Procedure

1 Open the **Profiles** tab of a virtual machine.

Option	Description
Edit the settings of a virtual machine	a Right-click a virtual machine from the inventory and select **Edit Settings**. b In the Virtual Machine Properties window, select the **Profiles** tab.
Use the virtual machine context menu	Right-click a virtual machine from the inventory and select **VM Storage Profile > Manage Profiles**.

2 Associate the virtual machine home files with a virtual machine storage profile from the **Home VM Storage Profile** drop-down menu.

NOTE The virtual machine home files include the file types `.vmx`, `.vmsd`, `.nvram`, and so on.

3 (Optional) Click **Propagate to disks** to associate all virtual disks with the same virtual machine storage profile.

4 Under VM storage profiles for virtual disks, associate each virtual disk with a different virtual machine storage profile from the **VM Storage Profile** drop-down menu.

5 Click **OK**.

The virtual machine storage profile name appears in the VM Storage Profiles pane of the **Summary** tab for the virtual machine.

NOTE If you add a new virtual disk and associate it with a virtual machine storage profile at the same time, the VMware vSphere Profile-Driven Storage Service might take some to associate the virtual machine storage profile with the new virtual disk.

Check Storage Compliance with Virtual Machine Storage Profile

When you associate a virtual machine storage profile with virtual machines and virtual disks, and select the datastores on which virtual machines and virtual disks run, you can check whether virtual machines and virtual disks use datastores that are compliant with their virtual machine storage profile.

If you check the compliance of a virtual machine whose host or cluster has virtual machine storage profiles disabled, the result of the check will be Non-compliant because the feature is disabled.

Prerequisites

To perform a compliance check for a virtual machine storage profile, you must associate the virtual machine storage profile with at least one virtual machine or virtual disk.

Procedure

1 In the vSphere Client, select **View > Management > VM Storage Profiles**.

2 Select a virtual machine storage profile from the inventory.

3 Select the **Virtual Machines** tab.

The **Virtual Machines** tab lists the virtual machines and the virtual disks that use the selected virtual machine storage profile.

4 Click **Check Compliance Now**.

The Compliance Status column shows whether the virtual machine files or the virtual disks use datastores that are compliant or noncompliant with the selected virtual machine storage profile.

Compliance Status	Description
Compliant	The datastore that the virtual machine or virtual disk uses has the storage capabilities that are required by the virtual machine storage profile.
Non-compliant	The datastore that the virtual machine or virtual disk uses does not have the storage capabilities that are required by the virtual machine storage profile. You can migrate the virtual machine files and virtual disks to compliant datastores.

What to do next

If a virtual machine or a virtual disk uses a datastore that is no longer compliant with the virtual machine storage profile, you can migrate it to a compliant datastore.

Check Storage Compliance for a Virtual Machine

You can check whether a virtual machine uses storage that is compliant with all virtual machine storage profiles that are associated with it.

Prerequisites

Before you check the profile compliance for a single virtual machine, ensure that the virtual machine has a virtual machine storage profile associated with it.

Procedure

1 From the vSphere Client inventory, right-click a virtual machine and select **VM Storage Profile > Check Profiles Compliance**.

2 Select the **Summary** tab.

3 View the compliance in the VM Storage Profiles pane beside the Profiles Compliance text box.

The Profiles Compliance text box in the VM Storage Profiles pane shows whether the virtual machine files or its virtual disks comply with their associated virtual machine storage profile.

Compliance Status	Description
Compliant	The datastores used by the virtual machine files or the virtual disks have the storage capabilities that are required by the respective virtual machine storage profile.
Non-compliant	The datastores used by the virtual machine or the virtual disks do not have the storage capabilities that are required by the respective virtual machine storage profile. You can migrate the virtual machine files and its virtual disks to a compliant datastore.

What to do next

If the status is noncompliant, read "Check Storage Compliance with Virtual Machine Storage Profile," on page 201 to view whether the virtual machine files or any of the virtual disks use noncompliant storage. You can then migrate the files or virtual disks to a compliant datastore.

Using vmkfstools

22

`vmkfstools` is one of the ESXi Shell commands for managing VMFS volumes and virtual disks. You can perform many storage operations using the `vmkfstools` command. For example, you can create and manage VMFS datastores on a physical partition, or manipulate virtual disk files, stored on VMFS or NFS datastores.

NOTE After you make a change using the `vmkfstools`, the vSphere Client may not be updated immediately. You need to use a refresh or rescan operation from the vSphere Client.

For more information on the ESXi Shell, see *Getting Started with vSphere Command-Line Interfaces*.

This chapter includes the following topics:

- "vmkfstools Command Syntax," on page 203
- "vmkfstools Options," on page 204

vmkfstools Command Syntax

Generally, you do not need to log in as the root user to run the `vmkfstools` commands. However, some commands, such as the file system commands, might require the root user login.

The `vmkfstools` command supports the following command syntax:

`vmkfstools` *conn_options options target*.

Target specifies a partition, device, or path to apply the command option to.

Table 22-1. `vmkfstools` command arguments

Argument	Description
options	One or more command-line options and associated arguments that you use to specify the activity for `vmkfstools` to perform, for example, choosing the disk format when creating a new virtual disk. After entering the option, specify a target on which to perform the operation. Target can indicate a partition, device, or path.
partition	Specifies disk partitions. This argument uses a *disk_ID:P* format, where *disk_ID* is the device ID returned by the storage array and *P* is an integer that represents the partition number. The partition digit must be greater than zero (0) and should correspond to a valid VMFS partition.

Table 22-1. `vmkfstools` command arguments (Continued)

Argument	Description
device	Specifies devices or logical volumes. This argument uses a path name in the ESXi device file system. The path name begins with `/vmfs/devices`, which is the mount point of the device file system. Use the following formats when you specify different types of devices: ■ `/vmfs/devices/disks` for local or SAN-based disks. ■ `/vmfs/devices/lvm` for ESXi logical volumes. ■ `/vmfs/devices/generic` for generic SCSI devices.
path	Specifies a VMFS file system or file. This argument is an absolute or relative path that names a directory symbolic link, a raw device mapping, or a file under `/vmfs`. ■ To specify a VMFS file system, use this format: `/vmfs/volumes/`*`file_system_UUID`* or `/vmfs/volumes/`*`file_system_label`* ■ To specify a file on a VMFS datastore, use this format: `/vmfs/volumes/`*`file_system_label`*`/` *`file_system_UUID`*`/[dir]/myDisk.vmdk` You do not need to enter the entire path if the current working directory is the parent directory of `myDisk.vmdk`.

vmkfstools Options

The `vmkfstools` command has several options. Some of the options are suggested for advanced users only.

The long and single-letter forms of the options are equivalent. For example, the following commands are identical.

```
vmkfstools --createfs vmfs5 --blocksize 1m disk_ID:P
vmkfstools -C vmfs5 -b 1m disk_ID:P
```

-v Suboption

The -v suboption indicates the verbosity level of the command output.

The format for this suboption is as follows:

```
-v --verbose number
```

You specify the *number* value as an integer from 1 through 10.

You can specify the -v suboption with any `vmkfstools` option. If the output of the option is not suitable for use with the -v suboption, `vmkfstools` ignores -v.

NOTE Because you can include the -v suboption in any `vmkfstools` command line, -v is not included as a suboption in the option descriptions.

File System Options

File system options allow you to create a VMFS file system. These options do not apply to NFS. You can perform many of these tasks through the vSphere Client.

Listing Attributes of a VMFS Volume

Use the `vmkfstools` command to list attributes of a VMFS volume.

```
-P --queryfs
      -h --human-readable
```

When you use this option on any file or directory that resides on a VMFS volume, the option lists the attributes of the specified volume. The listed attributes include the file system label, if any, the number of extents comprising the specified VMFS volume, the UUID, and a listing of the device names where each extent resides.

NOTE If any device backing VMFS file system goes offline, the number of extents and available space change accordingly.

You can specify the `-h` suboption with the `-P` option. If you do so, `vmkfstools` lists the capacity of the volume in a more readable form, for example, `5k`, `12.1M`, or `2.1G`.

Creating a VMFS File System

Use the `vmkfstools` command to create a VMFS datastore.

```
-C --createfs [vmfs3|vmfs5]
      -b --blocksize block_size kK|mM
      -S --setfsname datastore
```

This option creates a VMFS3 or VMFS5 datastore on the specified SCSI partition, such as *disk_ID:P*. The partition becomes the file system's head partition.

NOTE Use the VMFS3 option when you need legacy hosts to access the datastore.

You can specify the following suboptions with the `-C` option:

- `-b --blocksize` – Define the block size for the VMFS datastore.

  For VMFS5, the only available block size is 1MB. For VMFS3, the default block size is 1MB. Depending on your needs, the block size can be 1MB, 2MB, 4MB, and 8MB. When you enter the size, indicate the unit type by adding a suffix, such as m or M. The unit type is not case sensitive.

- `-S --setfsname` – Define the volume label of the VMFS datastore you are creating. Use this suboption only in conjunction with the `-C` option. The label you specify can be up to 128 characters long and cannot contain any leading or trailing blank spaces.

  After you define a volume label, you can use it whenever you specify the VMFS datastore for the `vmkfstools` command. The volume label appears in listings generated for the `ls -l` command and as a symbolic link to the VMFS volume under the `/vmfs/volumes` directory.

  To change the VMFS volume label, use the `ln -sf` command. Use the following as an example:

  ```
  ln -sf /vmfs/volumes/UUID /vmfs/volumes/datastore
  ```

  datastore is the new volume label to use for the *UUID* VMFS.

  NOTE If your host is registered with the vCenter Server, any changes you make to the VMFS volume label get overwritten by the vCenter Server. This guarantees that the VMFS label is consistent across all vCenter Server hosts.

Example for Creating a VMFS File System

This example illustrates creating a new VMFS datastore named my_vmfs on the `naa.`*`ID`*`:1` partition. The file block size is 1MB.

```
vmkfstools –C vmfs5 –b 1m –S my_vmfs /vmfs/devices/disks/naa.ID:1
```

Extending an Existing VMFS Volume

Use the `vmkfstools` command to add an extent to a VMFS volume.

`–Z ––spanfs` *`span_partition head_partition`*

This option extends the VMFS file system with the specified head partition by spanning it across the partition specified by *span_partition*. You must specify the full path name, for example `/vmfs/devices/disks/`*`disk_ID:1`*. Each time you use this option, you extend a VMFS volume with a new extent so that the volume spans multiple partitions.

Caution When you run this option, you lose all data that previously existed on the SCSI device you specified in *span_partition* .

Example for Extending a VMFS Volume

In this example, you extend the logical file system by allowing it to span to a new partition.

`vmkfstools –Z /vmfs/devices/disks/naa.`*`disk_ID_2`*`:1 /vmfs/devices/disks/naa.`*`disk_ID_1`*`:1`

The extended file system spans two partitions—`naa.`*`disk_ID_1`*`:1` and `naa.`*`disk_ID_2`*`:1`. In this example, `naa.`*`disk_ID_1`*`:1` is the name of the head partition.

Growing an Existing Extent

Instead of adding a new extent to a VMFS datastore, you can grow an existing extent using the `vmkfstools –G` command.

Use the following option to increase the size of a VMFS datastore after the underlying storage had its capacity increased.

`–G ––growfs` *`device device`*

This option grows an existing VMFS datastore or its extent. For example,

```
vmkfstools ––growfs /vmfs/devices/disks/disk_ID:1 /vmfs/devices/disks/disk_ID:1
```

Upgrading a VMFS Datastore

You can upgrade a VMFS3 to VMFS5 datastore.

Caution The upgrade is a one-way process. After you have converted a VMFS3 datastore to VMFS5, you cannot revert it back.

When upgrading the datastore, use the following command: `vmkfstools –T /vmfs/volumes/`*`UUID`*

Note All hosts accessing the datastore must support VMFS5 . If any ESX/ESXi host version 4.x or earlier is using the VMFS3 datastore, the upgrade fails and the host's mac address is displayed. with the Mac address details of the Host which is actively using the Datastore

Virtual Disk Options

Virtual disk options allow you to set up, migrate, and manage virtual disks stored in VMFS and NFS file systems. You can also perform most of these tasks through the vSphere Client.

Supported Disk Formats

When you create or clone a virtual disk, you can use the `-d --diskformat` suboption to specify the format for the disk.

Choose from the following formats:

- `zeroedthick` (default) – Space required for the virtual disk is allocated during creation. Any data remaining on the physical device is not erased during creation, but is zeroed out on demand at a later time on first write from the virtual machine. The virtual machine does not read stale data from disk.
- `eagerzeroedthick` – Space required for the virtual disk is allocated at creation time. In contrast to `zeroedthick` format, the data remaining on the physical device is zeroed out during creation. It might take much longer to create disks in this format than to create other types of disks.
- `thin` – Thin-provisioned virtual disk. Unlike with the `thick` format, space required for the virtual disk is not allocated during creation, but is supplied, zeroed out, on demand at a later time.
- `rdm:`*device* – Virtual compatibility mode raw disk mapping.
- `rdmp:`*device* – Physical compatibility mode (pass-through) raw disk mapping.
- `2gbsparse` – A sparse disk with 2GB maximum extent size. You can use disks in this format with hosted VMware products, such as VMware Fusion, Player, Server, or Workstation. However, you cannot power on sparse disk on an ESXi host unless you first re-import the disk with `vmkfstools` in a compatible format, such as `thick` or `thin`.

  See "Migrate Virtual Machines Between Different VMware Products," on page 209.

NFS Disk Formats

The only disk formats you can use for NFS are `thin`, `thick`, `zeroedthick` and `2gbsparse`.

`Thick`, `zeroedthick` and `thin` formats usually behave the same because the NFS server and not the ESXi host determines the allocation policy. The default allocation policy on most NFS servers is `thin`. However, on NFS servers that support Storage APIs - Array Integration, you can create virtual disks in `zeroedthick` format. The reserve space operation enables NFS servers to allocate and guarantee space.

For more information on array integration APIs, see Chapter 18, "Storage Hardware Acceleration," on page 173.

Creating a Virtual Disk

Use the `vmkfstools` command to create a virtual disk.

```
-c --createvirtualdisk size[kK|mM|gG]
      -a --adaptertype [buslogic|lsilogic|ide] srcfile
      -d --diskformat [thin|zeroedthick|eagerzeroedthick]
```

This option creates a virtual disk at the specified path on a datastore. Specify the size of the virtual disk. When you enter the value for *size*, you can indicate the unit type by adding a suffix of `k` (kilobytes), `m` (megabytes), or `g` (gigabytes). The unit type is not case sensitive. `vmkfstools` interprets either `k` or `K` to mean kilobytes. If you don't specify a unit type, `vmkfstools` defaults to bytes.

You can specify the following suboptions with the -c option.

- -a specifies the device driver that is used to communicate with the virtual disks. You can choose between BusLogic, LSI Logic, or IDE drivers.
- -d specifies disk formats.

Example for Creating a Virtual Disk

This example illustrates creating a two-gigabyte virtual disk file named rh6.2.vmdk on the VMFS file system named myVMFS. This file represents an empty virtual disk that virtual machines can access.

```
vmkfstools -c 2048m /vmfs/volumes/myVMFS/rh6.2.vmdk
```

Initializing a Virtual Disk

Use the vmkfstools command to initialize a virtual disk.

```
-w --writezeros
```

This option cleans the virtual disk by writing zeros over all its data. Depending on the size of your virtual disk and the I/O bandwidth to the device hosting the virtual disk, completing this command might take a long time.

CAUTION When you use this command, you lose any existing data on the virtual disk.

Inflating a Thin Virtual Disk

Use the vmkfstools command to inflate a thin virtual disk.

```
-j --inflatedisk
```

This option converts a thin virtual disk to eagerzeroedthick, preserving all existing data. The option allocates and zeroes out any blocks that are not already allocated.

Removing Zeroed Blocks

Use the vmkfstools command to convert any thin, zeroedthick, or eagerzeroedthick virtual disk to a thin disk with zeroed blocks removed.

```
-K --punchzero
```

This option deallocates all zeroed out blocks and leaves only those blocks that were allocated previously and contain valid data. The resulting virtual disk is in thin format.

Converting a Zeroedthick Virtual Disk to an Eagerzeroedthick Disk

Use the vmkfstools command to convert any zeroedthick virtual disk to an eagerzeroedthick disk.

```
-k --eagerzero
```

While performing the conversion, this option preserves any data on the virtual disk.

Deleting a Virtual Disk

This option deletes a virtual disk file at the specified path on the VMFS volume.

```
-U --deletevirtualdisk
```

Renaming a Virtual Disk

This option renames a virtual disk file at the specified path on the VMFS volume.

You must specify the original file name or file path *oldName* and the new file name or file path *newName*.

`–E ––renamevirtualdisk oldName newName`

Cloning a Virtual Disk or RDM

This option creates a copy of a virtual disk or raw disk you specify.

`–i ––clonevirtualdisk srcfile –d ––diskformat [rdm:device|rdmp:device|thin|2gbsparse]`

You can use the `–d` suboption for the `–i` option. This suboption specifies the disk format for the copy you create. A non-root user is not allowed to clone a virtual disk or an RDM.

Example for Cloning a Virtual Disk

This example illustrates cloning the contents of a master virtual disk from the `templates` repository to a virtual disk file named `myOS.vmdk` on the `myVMFS` file system.

```
vmkfstools –i /vmfs/volumes/myVMFS/templates/gold–master.vmdk /vmfs/volumes/myVMFS/myOS.vmdk
```

You can configure a virtual machine to use this virtual disk by adding lines to the virtual machine configuration file, as in the following example:

```
scsi0:0.present = TRUE
scsi0:0.fileName = /vmfs/volumes/myVMFS/myOS.vmdk
```

Migrate Virtual Machines Between DifferentVMware Products

Typically, you use VMware Converter to migrate virtual machines from other VMware products into your ESXi system. However, you can use the `vmkfstools –i` command to import virtual disks in 2gbsparse format into ESXi and then attach this disk to a new virtual machine you create in ESXi.

You must import the virtual disk first because you cannot power on disks in 2gbsparse format on the ESXi host.

Procedure

1 Import a disk in 2gbsparse format into the ESXi host by running the following command. Make sure to select the disk format compatible with ESXi.

```
vmkfstools –i <input> <output> –d <format>
```

2 In the vSphere Client, create a new virtual machine using the **Custom** configuration option.

3 When you configure a disk, select **Use an existing virtual disk** and attach the disk you imported.

Extending a Virtual Disk

This option extends the size of a disk allocated to a virtual machine after the virtual machine has been created.

`–X ––extendvirtualdisk newSize [kK|mM|gG]`

You must power off the virtual machine that uses this disk file before you enter this command. You might have to update the file system on the disk so the guest operating system can recognize and use the new size of the disk and take advantage of the extra space.

You specify the `newSize` parameter in kilobytes, megabytes, or gigabytes by adding a `k` (kilobytes), `m` (megabytes), or `g` (gigabytes) suffix. The unit type is not case sensitive. `vmkfstools` interprets either `k` or `K` to mean kilobytes. If you don't specify a unit type, `vmkfstools` defaults to kilobytes.

The `newSize` parameter defines the entire new size, not just the increment you add to the disk.

For example, to extend a 4g virtual disk by 1g, enter: `vmkfstools –X 5g` *disk name*.

You can extend the virtual disk to the eagerzeroedthick format by using the `–d eagerzeroedthick` option.

NOTE Do not extend the base disk of a virtual machine that has snapshots associated with it. If you do, you can no longer commit the snapshot or revert the base disk to its original size.

Upgrading Virtual Disks

This option converts the specified virtual disk file from ESX Server 2 format to the ESXi format.

`–M ––migratevirtualdisk`

Creating a Virtual Compatibility Mode Raw Device Mapping

This option creates a Raw Device Mapping (RDM) file on a VMFS volume and maps a raw LUN to this file. After this mapping is established, you can access the LUN as you would a normal VMFS virtual disk. The file length of the mapping is the same as the size of the raw LUN it points to.

`–r ––createrdm` *device*

When specifying the *device* parameter, use the following format:

`/vmfs/devices/disks/`*disk_ID:P*

Example for Creating a Virtual Compatibility Mode RDM

In this example, you create an RDM file named *my_rdm.vmdk* and map the *disk_ID* raw disk to that file.

`vmkfstools –r /vmfs/devices/disks/`*disk_ID* `my_rdm.vmdk`

You can configure a virtual machine to use the `my_rdm.vmdk` mapping file by adding the following lines to the virtual machine configuration file:

```
scsi0:0.present = TRUE
scsi0:0.fileName = /vmfs/volumes/myVMFS/my_rdm.vmdk
```

Creating a Physical Compatibility Mode Raw Device Mapping

This option lets you map a pass-through raw device to a file on a VMFS volume. This mapping lets a virtual machine bypass ESXi SCSI command filtering when accessing its virtual disk.This type of mapping is useful when the virtual machine needs to send proprietary SCSI commands, for example, when SAN-aware software runs on the virtual machine.

`–z ––createrdmpassthru` *device*

After you establish this type of mapping, you can use it to access the raw disk just as you would any other VMFS virtual disk.

When specifying the *device* parameter, use the following format:

`/vmfs/devices/disks/`*disk_ID*

Listing Attributes of an RDM

This option lets you list the attributes of a raw disk mapping.

`–q ––queryrdm`

This option prints the name of the raw disk RDM. The option also prints other identification information, like the disk ID, for the raw disk.

Displaying Virtual Disk Geometry

This option gets information about the geometry of a virtual disk.

`-g --geometry`

The output is in the form: `Geometry information C/H/S`, where `C` represents the number of cylinders, `H` represents the number of heads, and `S` represents the number of sectors.

NOTE When you import virtual disks from hosted VMware products to the ESXi host, you might see a disk geometry mismatch error message. A disk geometry mismatch might also be the cause of problems loading a guest operating system or running a newly-created virtual machine.

Checking and Repairing Virtual Disks

Use this option to check or repair a virtual disk in case of an unclean shutdown.

`-x , --fix [check|repair]`

Checking Disk Chain for Consistency

With this option, you can check the entire disk chain. You can determine if any of the links in the chain are corrupted or any invalid parent-child relationships exist.

`-e --chainConsistent`

Storage Device Options

Device options allows you to perform administrative task for physical storage devices.

Managing SCSI Reservations of LUNs

The `-L` option lets you reserve a SCSI LUN for exclusive use by the ESXi host, release a reservation so that other hosts can access the LUN, and reset a reservation, forcing all reservations from the target to be released.

`-L --lock [reserve|release|lunreset|targetreset|busreset]` *device*

CAUTION Using the `-L` option can interrupt the operations of other servers on a SAN. Use the `-L` option only when troubleshooting clustering setups.

Unless specifically advised by VMware, never use this option on a LUN hosting a VMFS volume.

You can specify the `-L` option in several ways:

- `-L reserve` – Reserves the specified LUN. After the reservation, only the server that reserved that LUN can access it. If other servers attempt to access that LUN, a reservation error results.
- `-L release` – Releases the reservation on the specified LUN. Other servers can access the LUN again.
- `-L lunreset` – Resets the specified LUN by clearing any reservation on the LUN and making the LUN available to all servers again. The reset does not affect any of the other LUNs on the device. If another LUN on the device is reserved, it remains reserved.
- `-L targetreset` – Resets the entire target. The reset clears any reservations on all the LUNs associated with that target and makes the LUNs available to all servers again.
- `-L busreset` – Resets all accessible targets on the bus. The reset clears any reservation on all the LUNs accessible through the bus and makes them available to all servers again.

When entering the *device* parameter, use the following format:

`/vmfs/devices/disks/`*disk_ID:P*

Breaking Device Locks

The `-B` option allows you to forcibly break the device lock on a particular partition.

`-B --breaklock` *`device`*

When entering the *device* parameter, use the following format:

`/vmfs/devices/disks/`*`disk_ID:P`*

You can use this command when a host fails in the middle of a datastore operation, such as grow extent, add extent, or resignaturing. When you issue this command, make sure that no other host is holding the lock.

Index

Symbols

* next to path **161**

A

access control **64**
accessing storage **20**
active-active disk arrays **32, 36, 47, 64, 68, 94, 163**
active-passive disk arrays, boot from SAN **52**
adaptive scheme **25**
add storage capability **196**
adding, NFS storage **128**
ADT, *See* auto disk transfer
advanced settings
  Disk.EnableNaviReg **58**
  Disk.MaxLUN **125**
alarms **192**
all-paths-down event **130**
allocations, LUN **36**
allocations, LUN **68**
applications, layered **27**
array integration, thin provisioning **186**
array-based solution **27**
assign storage capability **197, 198**
asterisk next to path **161**
atomic test and set **127**
ATS locking **127**
authentication **64, 82**
auto disk transfer **46**
auto volume transfer **46**
auto-detect,SSD devices **143**
automatic host registration, disabling **58**
AVT, *See* auto volume transfer

B

backups
  considerations **28**
  third-party backup package **29**
basic connectivity **43**
best practices, FCoE **37**
BIOS, enabling for BFS **54**
block devices **138**
boot adapters **53**
boot BIOS prompt, enabling for BFS **54**
boot from DVD-ROM **53**
boot from iSCSI SAN
  configuring HBAs **98**
  configuring iSCSI settings **99**
  guidelines **97**
  hardware iSCSI **98**
  iBFT **99**
  preparing SAN **98**
  software iSCSI **99**
boot from SAN
  benefits **51**
  boot LUN considerations **52**
  configuring Emulex HBAs **53**
  configuring Qlogic HBAs **55**
  configuring storage **52**
  HBA requirements **52**
  host requirements **52**
  overview **51**
  preparing installation **52**
  requirements **52**

C

CHAP
  disabling **85**
  for discovery targets **84**
  for iSCSI initiators **83**
  for static targets **84**
  mutual **82**
  one-way **82**
CHAP authentication **64, 82**
CHAP authentication methods **82**
checklist **111**
claim rules **161**
clustering **43**
compatibility modes
  physical **138**
  virtual **138**
configuring
  dynamic discovery **81**
  static discovery **82**
current multipathing state **162**

D

data digests **65**
datastore, storage capability **197, 198**
datastores
  adding extents **119**

configuring on NFS volumes **128**
displaying **19**
grouping **129**
increasing capacity **119**
managing duplicate **122**
mounting **122**
NFS **113**
paths **162**
refreshing **124**
renaming **129**
review properties **20**
storage over-subscription **186**
unmounting **128**
VMFS **113**
Dell PowerVault MD3000i storage systems **96**
dependent hardware iSCSI
and associated NICs **72**
configuration workflow **70**
considerations **71**
dependent iSCSI, networking **74**
device locks, breaking **212**
device loss, unplanned **132**
diagnostic partition, configuring **133**
diagnostic partitions **35, 68**
direct connect **43**
disabling paths **163**
disaster recovery **24**
discovery
address **81**
dynamic **81**
static **82**
disk arrays
active-active **36, 68, 163**
active-passive **36, 68, 163**
disk chain, consistency **211**
disk formats
thick provisioned **184**
thin provisioned **184**
disk mirroring **134**
disk shares **26**
disk timeout **157**
Disk.EnableNaviReg **58**
Disk.MaxLUN **125**
disks
format **185**
inflate **186**
dump partitions **35, 68**
DVD-ROM, booting from **53**
dynamic discovery, configuring **81**
dynamic discovery addresses **81**
dynamic disks **134**

E

educational support **7**
EMC CLARiiON **44, 92**
EMC CLARiiON AX100
and RDM **45**
directly connected **45**
EMC Symmetrix, pseudo LUNs **45, 93**
EqualLogic, storage systems **95**
ESXi, configuring for net dump **105**
EUI **62**
EVA (HP StorageWorks) **48, 94**
events **192**
examples
vmkfstools -C **206**
vmkfstools -Z **206**
extents
adding to datastore **119**
growing **119**

F

failover
I/O delay **156**
transparent **32, 64**
failover paths, status **161**
FC HBA setup **36**
FC SAN
accessing **33**
hardware requirements **35**
FCoE, best practices **37**
FCoE adapters **37**
Fibre Channel
concepts **31**
configuration checklist **59**
Fibre Channel SAN
best practices **57**
preventing problems **57**
file systems, upgrading **114**
file-based (VMFS) solution **27**
FIP **37**
Fixed path policy **160, 163**

G

GPT **16**

H

hardware acceleration
about **173**
benefits **173**
block storage **174**
deleting claim rules **178**
enabling **175**
NAS **179**
NAS status **181**
requirements **174**

status **174**
support details **176**
hardware acceleration, considerations **181**
hardware iSCSI, and failover **154**
hardware iSCSI adapters
dependent **63**
independent **63**
hardware iSCSI initiators
configuring **69**
installing **69**
setting up discovery addresses **81**
viewing **69**
HBAs
queue depth **35**
setup **36**
static load balancing **36**
header digests **65**
high-tier storage **26**
Hitachi Data Systems storage, microcode **48**
host configuration, advanced settings **58**
host registration, disabling **58**
host type **44, 92**
host-based failover **153**
hosts, and FC SAN **31**
HP LeftHand P4000 VSA **96**
HP StorageWorks
EVA **48, 94**
MSA **93**
XP **48**
HP StorageWorks SAN/iQ storage **95**

I

I/O delay **68, 156**
iBFT **99**
iBFT iSCSI boot
booting an ESXi host **102**
changing boot sequence **101**
installing an ESXi host **102**
limitations **100**
networking best practices **102**
setting up ESXi **100**
troubleshooting **103**
IBM ESS800 **47**
IBM FAStT **46**
IBM System Storage DS4800, failover configuration **46**
IBM Systems Storage 8000 **47**
IDE **11**
independent hardware iSCSI adapters
change IP address **70**
change name **70**
installation
preparing for boot from SAN **52**
steps **36**
inter-switch link **46**
IP address **62**
IQN **62**
iSCSI **12**
iSCSI adapters
about **67**
advanced parameters **86**
hardware **63**
software **63**
iSCSI alias **62**
iSCSI boot, iBFT **99**
iSCSI Boot Firmware Table, *See* iBFT
iSCSI boot parameters, configuring **101**
iSCSI initiators
configuring advanced parameters **87**
configuring CHAP **83**
hardware **69**
setting up CHAP parameters **82**
iSCSI names, conventions **62**
iSCSI networking
binding adapters **78**
changing policy **78**
creating a VMkernel interface **76**
managing **79**
troubleshooting **79**
iSCSI ports **62**
iSCSI SAN
accessing **66**
best practices **107**
boot **97**
concepts **61**
preventing problems **107**
iSCSI SAN restrictions **68**
iSCSI sessions
adding for a target **88**
displaying **88**
managing **87**
removing **89**
iSCSI storage systems **91**
ISL **46**

J

jumbo frames
enabling for dependent hardware iSCSI **80**
enabling for software iSCSI **80**
using with iSCSI **80**

L

layered applications **27**
Linux, host type **44**

Linux Cluster, host type **44**
Linux Cluster host type **92**
Linux host type **92**
load balancing **24, 36**
locations of virtual machines **26**
loss of network connection, troubleshooting **103**
lower-tier storage **26**
LUN decisions
  adaptive scheme **25**
  predictive scheme **25**
LUN masking **31**
LUNs
  allocations **36, 68**
  and VMFS datastores **35**
  changing number scanned **125**
  decisions **24**
  making changes and rescan **124**
  masking **169**
  multipathing policy **163**
  NPIV-based access **39**
  one VMFS volume per **68**
  setting multipathing policy **163**

M

maintenance **24**
masking LUNs **169**
MBR **16**
metadata, RDMs **138**
metadata updates **116**
microcode, Hitachi Data Systems storage **48**
Microsoft Cluster Service **43**
mid-tier storage **26**
Most Recently Used path policy **160, 163**
mounting VMFS datastores **131**
MPPs
  displaying **166**
  See also multipathing plug-ins
MRU path policy **163**
MSA (HP StorageWorks) **93**
MSCS **43**
multipathing
  active paths **161**
  broken paths **161**
  considerations **164**
  disabled paths **161**
  standby paths **161**
  viewing the current state of **161**
multipathing claim rules
  adding **167**
  deleting **168**
multipathing plug-ins, path claiming **161**
multipathing policy **163**
multipathing state **162**
mutual CHAP **82**

N

N-Port ID Virtualization, *See* NPIV
NAA **62**
NAS **12**
NAS plug-ins
  installing **179**
  uninstalling **180**
  upgrading **180**
Native Multipathing Plug-In **158, 159**
net dump
  configuring ESXi **105**
  configuring vMA **104**
NetApp storage system **95**
Netware host mode **48**
network adapters, configuring for iBFT iSCSI boot **101**
Network Appliance storage **48**
network connections, create **76**
network performance **109**
networking, configuring **69**
NFS datastores
  and non-ASCII characters **127**
  maximum size **127**
  repositories **127**
  unmounting **128**
NFS storage, adding **128**
NICs, mapping to VMkernel **77**
NMP
  I/O flow **160**
  path claiming **161**
  See also Native Multipathing Plug-In
NPIV
  about **39**
  assigning WWNs **41**
  changing WWNs **41**
  limitations **40**
  requirements **40**

O

one-way CHAP **82**

P

partition mappings **138**
passive disk arrays **36, 68, 163**
path claiming **161**
path failover
  and virtual machines **157**
  array-based **156**
  host-based **154**
path failure rescan **124**

path management **153**
path policies
  changing defaults **163**
  Fixed **156, 160, 163**
  Most Recently Used **160, 163**
  MRU **163**
  Round Robin **160, 163**
Path Selection Plug-Ins **160**
path thrashing **44, 92**
paths
  disabling **163**
  masking **169**
  preferred **161**
  unmasking **170**
performance
  checking Ethernet switch statistics **111**
  network **109**
  optimizing **58, 108**
  storage system **108**
permanent device loss **130**
planned device removal **130**
Pluggable Storage Architecture **158**
port binding **154**
port redirection **156**
Port_ID **32**
predictive scheme **25**
preferred path **161**
prioritizing virtual machines **26**
profile-driven storage **195**
PSA, *See* Pluggable Storage Architecture
PSPs, *See* Path Selection Plug-Ins

Q

Qlogic HBA BIOS, enabling for BFS **55**
queue depth **68**

R

RAID devices **138**
raw device mapping, see RDM **135**
RDM
  advantages **136**
  and virtual disk files **139**
  dynamic name resolution **139**
  overview **135**
  physical compatibility mode **138**
  virtual compatibility mode **138**
  with clustering **139**
RDMs
  and snapshots **138**
  path management **141**
reclaiming space **189**
remove a storage capability **198**
requirements, boot from SAN **52**
rescan
  LUN creation **124**
  path masking **124**
  when path is down **124**
rescanning
  datastores **124**
  storage adapters **124**
  storage devices **124**
resignaturing **122**
restrictions **35**
Round Robin path policy **160, 163**

S

SAN
  backup considerations **28**
  benefits **23**
  hardware failover **46**
  requirements **35**
  specifics **24**
SAN fabric **31**
SAN management software **28**
SAN storage performance, optimizing **58, 108**
SAS **11**
SATA **11**
SATPs
  adding rules **171**
  displaying **166**
  See also Storage Array Type Plug-Ins
scanning, changing number **125**
SCSI, vmkfstools **203**
SCSI controllers **9**
server performance **59, 108**
setup steps **36**
software FCoE
  and VMkernel **38**
  activating adapters **39**
software iSCSI
  and failover **154**
  diagnostic partition **133**
  networking **74**
software iSCSI adapter
  configuring **72**
  disabling **73**
software iSCSI boot, changing settings **103**
software iSCSI initiator, enabling **73**
software iSCSI initiators, setting up discovery addresses **81**
SSD **143**
SSD devices
  auto-detect **143**
  tag **144**
SSD enablement, benefits **143**
standard switches **77**

static discovery, configuring **82**
static discovery addresses **81**
storage
  access for virtual machines **20**
  adapters **10**
  introduction **9**
  local **11**
  networked **12**
  not-shared **185**
  provisioned **185**
  provisioning **183**
  supported vSphere features **21**
  types **11**
  used by virtual machines **185**
storage adapters
  about **70**
  copying names **11**
  rescanning **124**
  viewing **10**
  viewing in vSphere Client **10**
Storage APIs, Storage Awareness **191**
storage area network **61**
Storage Array Type Plug-Ins **159**
storage arrays
  configuring **43**
  LSI-based **49**
  performance **58**
storage capabilities **191**
storage capability
  add **196**
  assign to datastore **197**
  assign to multile datastores **198**
  definition **195**
  edit **197**
  remove **198**
  system-defined **196**
  user-defined **196**
  view **196**
  view existing **196**
storage compliance
  per virtual machine **202**
  per virtual machine storage profile **201**
storage device, connection status **132**
storage devices
  attaching **131**
  detaching **130**
  disconnections **130**
  displaying **166**
  displaying for a host **18**
  displaying for an adapter **18**
  hardware acceleration status **176**
  identifiers **18**
  naming **17**
  paths **162**
  rescanning **124**
  viewing **16**
storage filters
  disabling **125**
  host rescan **126**
  RDM **126**
  same host and transports **126**
  VMFS **126**
storage processors
  configuring sense data **47**
  sense data **47**
storage space **183**
storage status **191**
storage systems
  Dell PowerVault MD3000i **96**
  EMC CLARiiON **44, 92**
  EMC Symmetrix **45, 93**
  EqualLogic **95**
  Hitachi **48**
  HP StorageWorks **47, 93**
  HP StorageWorks SAN/iQ **95**
  NetApp **95**
  Network Appliance **48**
  performance **108**
  types **32, 64**
storage topology **191**
storage virtualization **9**
storage, and VMkernel **149**
STP **37**
supported devices **44**
system-defined storage capability **196**

T

tag, SSD devices **144**
tag devices **146**
tape devices **36**
targets **15, 63**
targets vs. LUNs **63**
technical support **7**
testing, storage systems **91**
thin disks, creating **184**
thin provisioned LUNs, space reclamation **188**
thin provisioning, over-subscription **183**
thin-provisioned LUNs
  identify **187**
  reporting **187**
third-party backup package **29**
third-party management applications **28**
TimeoutValue parameter **35, 68**
troubleshooting
  changing iSCSI boot parameters **104**
  loss of network connection **103**

U

unplanned device loss **132**
untag **145, 146**
USB **11**
use cases **24**
user-defined storage capability **196**

V

VAAI claim rules
  defining **177**
  deleting **178**
  VAAI filter **177**
  VAAI plug-in **177**
VAAI filter, displaying **175**
VAAI plug-ins, displaying **175**
vendor provider, requirements **192**
vendor providers
  registering **193**
  SSL connection **193**
  unregistering **194**
  updating **194**
  viewing **193**
view storage capabilities **196**
virtual disk, repair **211**
virtual disks
  extending **209**
  formats **184**
  supported formats **207**
virtual machine storage profile
  associate with virtual disks **201**
  associate with virtual machine **201**
  compliance **198, 201, 202**
  create **199**
  definition **198**
  delete **200**
  edit **200**
  enable **199**
virtual machines
  accessing FC SAN **33**
  accessing iSCSI SAN **66**
  assigning WWNs to **41**
  I/O delay **156**
  locations **26**
  prioritizing **26**
  with RDMs **140**
virtual ports (VPORTs) **39**
virtual SSD device **148**
vMA, collecting net dump **104**
vMA, configuring for net dump **104**
VMFS
  conversion **206**
  locking **117**
  one volume per LUN **68**
  resignaturing **122**
  vmkfstools **203**
VMFS datastores
  adding extents **119**
  changing properties **119**
  changing signatures **123**
  creating **117**
  creating on Fibre Channel storage **118**
  creating on iSCSI storage **118**
  creating on SCSI disk **118**
  creating on storage device **118**
  deleting **123**
  disk formats **115**
  increasing capacity **119**
  mounting **131**
  rescanning **124**
  sharing **116**
  unmounting **128**
  upgrading **120**
VMFS resignaturing **122**
VMFS2 datastores, upgrading **121**
VMFS3, upgrading **121**
VMFS5, improvements over VMFS3 **114**
VMFS5 and VMFS3, differences **114**
VMkernel interfaces **77**
vmkfstools
  breaking locks **212**
  cloning disks **209**
  creating RDMs **210**
  creating virtual disks **207**
  deleting virtual disks **208**
  device options **211**
  disk chain **211**
  extending virtual disks **209**
  file system options **205**
  geometry **211**
  inflating thin disks **208**
  initializing virtual disks **208**
  migrating virtual disks **209**
  overview **203**
  RDM attributes **210**
  removing zeroed blocks **208**
  renaming virtual disks **209**
  SCSI reservations **211**
  syntax **203**
  upgrading virtual disks **210**
  virtual disk options **207**
  virtual disks conversion **208**
vmkfstools -C command **205**
vmkfstools -G command **206**
vmkfstools -P command **205**
vmkfstools -v command **204**

vmkfstools -Z command **206**
vmkfstools command options **204**
vmkfstools examples
  cloning disks **209**
  creating RDMs **210**
  creating virtual disks **208**
vMotion **23, 24, 36, 68, 92**
vmware, host type **44**
VMware DRS, using with vMotion **68**
VMware HA **23, 92**
VMware NMP
  I/O flow **160**
  See also Native Multipathing Plug-In
VMware PSPs, *See* Path Selection Plug-Ins
VMware SATPs, *See* Storage Array Type Plug-Ins

W

Windows guest OS timeout **157**
World Wide Names, *See* WWNs
World Wide Port Names, *See* WWPNs
WWNNs **41**
WWNs
  assigning to virtual machines **41**
  changing **41**
WWPNs **32, 41**

X

XP (HP StorageWorks) **48**

Z

zoning **31, 32**