

Hardware Information

Using the Virtual I/O Server

ESCALA POWER5



REFERENCE
86 A1 24EW 00

ESCALA POWER5

Hardware Information

Using the Virtual I/O Server

Hardware

July 2006

BULL CEDOC

357 AVENUE PATTON

B.P.20845

49008 ANGERS CEDEX 01

FRANCE

REFERENCE

86 A1 24EW 00

The following copyright notice protects this book under Copyright laws which prohibit such actions as, but not limited to, copying, distributing, modifying, and making derivative works.

Copyright © Bull SAS 1992, 2006

Printed in France

Suggestions and criticisms concerning the form, content, and presentation of this book are invited. A form is provided at the end of this book for this purpose.

To order additional copies of this book or other Bull Technical Publications, you are invited to use the Ordering Form also provided at the end of this book.

Trademarks and Acknowledgements

We acknowledge the right of proprietors of trademarks mentioned in this book.

AIX® is a registered trademark of International Business Machines Corporation, and is being used under licence.

UNIX® is a registered trademark in the United States of America and other countries licensed exclusively through the Open Group.

Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries

Table of Contents

Using the Virtual I/O Server.....	1
Printable PDFs.....	1
Virtual I/O Server overview.....	2
Virtual SCSI.....	2
Concepts for the Virtual I/O Server.....	3
Concepts for Virtual SCSI.....	3
Concepts for virtual networking.....	11
Virtual I/O Server management.....	13
Planning for the Virtual I/O Server.....	17
Specifications.....	17
Limitations and restrictions.....	18
Capacity planning.....	19
Redundancy considerations.....	29
Security considerations.....	33
Installing the Virtual I/O Server.....	33
Installing the Virtual I/O Server in an HMC environment.....	34
Installing the Virtual I/O Server in a non-HMC environment.....	39
Connecting to the Virtual I/O Server using OpenSSH.....	39
Configuration scenarios for the Virtual I/O Server.....	41
Scenario: Configuring a Virtual I/O Server without VLAN tagging.....	41
Scenario: Configuring a Virtual I/O Server using VLAN tagging.....	42
Scenario: Configuring shared Ethernet adapter failover.....	44
Scenario: Configuring Network Interface Backup in Virtual I/O clients without VLAN tagging.....	45
Scenario: Configuring Multi-Path I/O for AIX client logical partitions.....	47
Securing the Virtual I/O Server.....	49
Managing the Virtual I/O Server.....	52
Managing shared Ethernet adapters.....	52
Managing Virtual SCSI.....	63
Maintaining the Virtual I/O Server.....	71
Troubleshooting the Virtual I/O Server.....	77
Troubleshooting the Virtual I/O Server logical partition.....	77
Troubleshooting the client logical partition.....	81
Related information for the Virtual I/O Server.....	84
Virtual I/O Server command descriptions.....	84
Installation commands.....	85

Using the Virtual I/O Server

The purpose of this information is to familiarize you with the Virtual I/O Server, to help you plan for the Virtual I/O Server in your computing environment, and to give you configuration and management instructions.

- **Printable PDFs**
If you prefer a hardcopy version of this information, go here to print the PDF. Links to PDF documents about related topics are also included here.
 - **Virtual I/O Server overview**
Learn the concepts of the Virtual I/O Server and its primary components.
 - **Concepts for the Virtual I/O Server**
Become familiar with the Virtual I/O Server concepts, including the command-line interface, user types, virtual networking, and virtual SCSI.
 - **Planning for the Virtual I/O Server**
Use this topic to help gain an understanding of what to consider when planning for the Virtual I/O Server. In this section, you will find information about planning for the Virtual I/O Server.
 - **Installing the Virtual I/O Server**
Find instructions for installing the Virtual I/O Server.
 - **Configuration scenarios for the Virtual I/O Server**
The following scenarios show examples of networking configurations for the Virtual I/O Server logical partition and the client logical partitions. Use the following scenarios and configuration examples to understand more about the Virtual I/O Server and its components.
 - **Managing the Virtual I/O Server**
Find information about managing Virtual I/O Server user types, adding and removing physical resources, and managing logical volumes. Also find information about backing up, restoring, updating, and monitoring the Virtual I/O Server.
 - **Troubleshooting the Virtual I/O Server**
Find information about diagnosing Virtual I/O Server problems and information about how to correct those problems.
 - **Related information for the Virtual I/O Server**
Find other information related to the Virtual I/O Server.
 - **Virtual I/O Server command descriptions**
This topic includes descriptions of the Virtual I/O Server commands.
-

Printable PDFs

If you prefer a hardcopy version of this information, go here to print the PDF. Links to PDF documents about related topics are also included here.

To view or download the PDF version of this document, select [Using the Virtual I/O Server](#).

You can view or download these related topics:

- [Creating a virtual computing environment](#)
- [Virtual I/O Server Commands Reference](#)
- [Partitioning with Integrated Virtualization Manager](#)
- [Managing the Integrated Virtualization Manager](#)
- [Partitioning for AIX](#)
- [Partitioning for Linux](#)

Saving PDF files

To save a PDF on your workstation for viewing or printing:

1. Right-click the PDF in your browser (right-click the link above).
2. Click Save Target As... if you are using Internet Explorer. Click Save Link As... if you are using Netscape Communicator.
3. Navigate to the directory in which you would like to save the PDF.
4. Click Save.

Downloading Adobe Acrobat Reader

You need Adobe Acrobat Reader to view or print these PDFs. You can download a copy from the [Adobe Web site](http://www.adobe.com/products/acrobat/readstep.html) (www.adobe.com/products/acrobat/readstep.html).

Parent topic: [Using the Virtual I/O Server](#)

Virtual I/O Server overview

Learn the concepts of the Virtual I/O Server and its primary components.

The Virtual I/O Server is software that resides in a POWER5 logical partition. This software facilitates the sharing of physical I/O resources between AIX and Linux client logical partitions within the server. The Virtual I/O Server provides virtual SCSI target and shared Ethernet adapter capability to client logical partitions within the system, allowing the client logical partitions to share SCSI devices and Ethernet adapters. The Virtual I/O Server software requires that the logical partition be dedicated solely for its use.

The Virtual I/O Server is available as part of the Advanced POWER Virtualization hardware feature.

Using the Virtual I/O Server facilitates the following functions:

- Sharing of physical resources between partitions on the system
- Creating partitions without requiring additional physical I/O resources
- Creating more partitions than there are I/O slots or physical devices available with the ability for partitions to have dedicated I/O, virtual I/O, or both
- Maximizing physical resource use on the system
- Helping to reduce the Storage Area Network (SAN) infrastructure

The Virtual I/O Server supports client logical partitions running the following operating systems:

- AIX 5.3 and later.
- SUSE LINUX Enterprise Server 9 for POWER
- Red Hat Enterprise Linux AS for POWER Version 3
- Red Hat Enterprise Linux AS for POWER Version 4

The Virtual I/O Server is comprised of the following primary components:

- Virtual SCSI
- Virtual Networking
- Integrated Virtualization Manager

The following sections provide a brief overview of each of these components.

Virtual SCSI

Physical adapters with attached disks or optical devices on the Virtual I/O Server logical partition can be shared by one or more client logical partitions. The Virtual I/O Server offers a storage subsystem that provides standard SCSI-compliant logical unit numbers (LUNs). The Virtual I/O Server is capable of exporting a pool of heterogeneous physical storage as an homogeneous pool of block storage in the form of SCSI disks. The Virtual I/O Server is a localized storage subsystem.

Unlike typical storage subsystems that are physically located out in the SAN, the SCSI devices exported by the Virtual I/O Server are limited to the domain within the server. Although the SCSI LUNs are SCSI compliant, they might not meet the needs of all applications, particularly those that exist in a distributed environment.

The following SCSI peripheral-device types are supported:

- Disks backed by a logical volume
- Disks backed by a physical volume
- Optical devices (DVD-RAM and DVD-ROM)

For more information about virtual SCSI, see [Concepts for Virtual SCSI](#).

Virtual networking

Shared Ethernet adapter allows logical partitions on the virtual local area network (VLAN) to share access to a physical Ethernet adapter and to communicate with systems and partitions outside the server. This function enables logical partitions on the internal VLAN to share the VLAN with standalone servers.

For more information about virtual networking, see [Concepts for virtual networking](#).

Integrated Virtualization Manager

The Integrated Virtualization Manager provides a browser-based interface and a command-line interface that you can use to manage servers that use the Virtual I/O Server. On the managed system, you can create logical partitions, manage the virtual storage and virtual Ethernet, and view service information related to the server. The Integrated Virtualization Manager is packaged with the Virtual I/O Server, but it is activated and usable only on certain platforms and where no Hardware Management Console (HMC) is present.

For more information about the Integrated Virtualization Manager, see [Managing the Integrated Virtualization Manager](#) and [Partitioning with the Integrated Virtualization Manager](#).

Parent topic: [Using the Virtual I/O Server](#)

Concepts for the Virtual I/O Server

Become familiar with the Virtual I/O Server concepts, including the command-line interface, user types, virtual networking, and virtual SCSI.

- [Concepts for Virtual SCSI](#)
Virtual SCSI allows client logical partitions to share disk storage and optical devices that are assigned to the Virtual I/O Server logical partition.
- [Concepts for virtual networking](#)
Use this section to find information about virtual Ethernet, shared Ethernet adapter, shared Ethernet adapter failover, link aggregation, and VLAN.
- [Virtual I/O Server management](#)
This topic contains information about Virtual I/O Server management interfaces, such as the Virtual I/O Server command-line interface and the Integrated Virtualization Manager. Virtual I/O Server user types are also explained

Parent topic: [Using the Virtual I/O Server](#)

Concepts for Virtual SCSI

Virtual SCSI allows client logical partitions to share disk storage and optical devices that are assigned to the Virtual I/O Server logical partition.

Disks and optical devices attached to physical adapter in the Virtual I/O Server logical partition can be shared by one or more client logical partitions. The Virtual I/O Server is a standard storage subsystem that provides

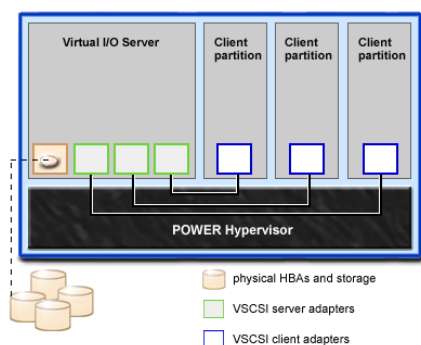
standard SCSI-compliant LUNs. The Virtual I/O Server is capable of exporting a pool of heterogeneous physical storage as a homogeneous pool of block storage in the form of SCSI disks. The Virtual I/O Server is a localized storage subsystem. Unlike typical storage subsystems that are physically located in the SAN, the SCSI devices exported by the Virtual I/O Server are limited to the domain within the server. Therefore, although the SCSI LUNs are SCSI compliant, they might not meet the needs of all applications, particularly those that exist in a distributed environment.

The following SCSI peripheral device types are supported:

- Disk backed by logical volume
- Disk backed by physical volume
- Optical CD-ROM, DVD-RAM, and DVD-ROM

Virtual SCSI is based on a client-server relationship. The Virtual I/O Server owns the physical resources as well as the *virtual SCSI server adapter*, and acts as a server, or SCSI target device. The client logical partitions have a SCSI initiator referred to as the *virtual SCSI client adapter*, and access the virtual SCSI targets as standard SCSI LUNs. You configure the virtual adapters by using the HMC or Integrated Virtualization Manager. The configuration and provisioning of virtual disk resources is performed by using the Virtual I/O Server. Physical disks owned by the Virtual I/O Server can be either exported and assigned to a client logical partition as a whole or can be partitioned into logical volumes. The logical volumes can then be assigned to different partitions. Therefore, virtual SCSI enables the sharing of adapters as well as disk devices. To make a physical or a logical volume available to a client partition requires that it be assigned to a virtual SCSI server adapter on the Virtual I/O Server. The client logical partition accesses its assigned disks through a virtual-SCSI client adapter. The virtual-SCSI client adapter recognizes standard SCSI devices and LUNs through this virtual adapter.

The following figure shows a standard virtual SCSI configuration.



Note: In order for client partitions to be able to access virtual devices, the Virtual I/O Server must be fully operational.

- [Virtual I/O Server storage subsystem overview](#)
Learn about the Virtual I/O Server storage subsystem.
- [Concepts for physical storage](#)
Learn more about physical storage, logical volumes, and the devices and configurations that are supported by the Virtual I/O Server.
- [Concepts for virtual storage](#)
Disks and optical devices are supported as virtual SCSI devices. This topic describes how those devices function in a virtualized environment and provides information on what devices are supported.
- [Concepts for mapping devices](#)
Mapping devices are used to facilitate the mapping of physical resources to a virtual device.

Parent topic: [Concepts for the Virtual I/O Server](#)

Virtual I/O Server storage subsystem overview

Learn about the Virtual I/O Server storage subsystem.

The Virtual I/O Server storage subsystem is a standard storage subsystem that provides standard SCSI-compliant LUNs. The Virtual I/O Server is a localized storage subsystem. Unlike typical storage subsystems that are physically located in the SAN, the SCSI devices exported by the Virtual I/O Server are limited to the domain within the server. Although the SCSI LUNs are SCSI-compliant, they might not meet the needs of all applications, particularly those that exist in a distributed environment.

Like typical disk storage subsystems, the Virtual I/O Server has a distinct front end and back end. The front end is the interface to which client logical partitions attach to view standard SCSI-compliant LUNs. Devices on the front end are called *virtual SCSI devices*. The back end is made up of physical storage resources. These physical resources include physical disk storage, both SAN devices and internal storage devices, optical devices, and logical volumes.

To create a virtual device, some physical storage must be allocated and assigned to a virtual SCSI server adapter. This process creates a virtual device instance (vtscsiX). The device instance can be considered a mapping device. It is not a real device, but rather a mechanism for managing the mapping of the portion of physical back-end storage to the front-end virtual SCSI device. This mapping device is instrumental in recreating the physical-to-virtual allocations in a persistent manner when the Virtual I/O Server is restarted.

Parent topic: [Concepts for Virtual SCSI](#)

Concepts for physical storage

Learn more about physical storage, logical volumes, and the devices and configurations that are supported by the Virtual I/O Server.

- **Physical volumes**
Physical volumes might be exported to client partitions as virtual SCSI disks. The Virtual I/O Server is capable of taking a pool of heterogeneous physical disk storage attached to its back end and exporting this as homogeneous storage in the form of SCSI disk LUNs.
- **Logical volumes**
Understand how logical volumes can be exported to client partitions as virtual SCSI disks. A logical volume is a portion of a physical volume.
- **Optical devices**
Optical devices can be exported by the Virtual I/O Server. This topic gives information about what types of optical devices are supported.

Parent topic: [Concepts for Virtual SCSI](#)

Physical volumes

Physical volumes might be exported to client partitions as virtual SCSI disks. The Virtual I/O Server is capable of taking a pool of heterogeneous physical disk storage attached to its back end and exporting this as homogeneous storage in the form of SCSI disk LUNs.

The Virtual I/O Server must be able to accurately identify a physical volume each time it boots, even if an event such as a storage area network (SAN) reconfiguration or adapter change has taken place. Physical volume attributes, such as the name, address, and location, might change after the system reboots due to SAN reconfiguration. However, the Virtual I/O Server must be able to recognize that this is the same device and update the virtual device mappings. For this reason, in order to export a physical volume as a virtual device, the physical volume must have either a unique identifier (UDID), a physical identifier (PVID), or an IEEE volume attribute.

For instructions on how to determine whether your disks have one of these identifiers, see [Identifying exportable disks](#).

The following commands are used to manage physical volumes:

Table 1. Physical volume commands and their descriptions

Physical volume command	Description
lspv	Displays information about a physical volume within a volume group.
migratepv	Moves allocated physical partitions from one physical volume to one or more other physical volumes.

Parent topic: [Concepts for physical storage](#)

Logical volumes

Understand how logical volumes can be exported to client partitions as virtual SCSI disks. A logical volume is a portion of a physical volume.

A hierarchy of structures is used to manage disk storage. Each individual disk drive or LUN, called a *physical volume*, has a name, such as `/dev/hdisk0`. Every physical volume in use either belongs to a volume group or is used directly for virtual storage. All of the physical volumes in a volume group are divided into physical partitions of the same size. The number of physical partitions in each region varies, depending on the total capacity of the disk drive.

Within each volume group, one or more logical volumes are defined. Logical volumes are groups of information located on physical volumes. Data on logical volumes appears to the user to be contiguous but can be discontinuous on the physical volume. This allows logical volumes to be resized or relocated and to have their contents replicated.

Each logical volume consists of one or more logical partitions. Each logical partition corresponds to at least one physical partition. Although the logical partitions are numbered consecutively, the underlying physical partitions are not necessarily consecutive or contiguous.

After installation, the system has one volume group (the `rootvg` volume group) consisting of a base set of logical volumes required to start the system. For more information on volume groups, see [Volume groups](#).

You can use the commands described in the following table to manage logical volumes.

Table 1. Logical volume commands and their descriptions

Logical volume command	Description
chlv	Changes the characteristics of a logical volume.
cplv	Copies the contents of a logical volume to a new logical volume.
extendlv	Increases the size of a logical volume.
lslv	Displays information about the logical volume.
mklv	Creates a logical volume.
mklvcopy	Creates a mirror of a logical volume.
rmlv	Removes logical volumes from a volume group.
rmlvcopy	Removes a copy of a logical volume.

Creating one or more distinct volume groups rather than using logical volumes that are created in the `rootvg` volume group allows you to install any newer versions of the Virtual I/O Server while maintaining client data by exporting and importing the volume groups created for virtual I/O.

Note: Logical volumes used as virtual disks must be less than 1 TB in size. In addition, logical volumes on the Virtual I/O Server that are going to be used as virtual disks cannot be mirrored, striped, or have bad block relocation enabled.

- **Volume groups**
Find information about volume groups.
- **Physical partitions**
This topic contains information about physical partitions.
- **Logical partitions**
This topic contains information logical storage partitions.
- **Quorums**
Find information about quorums.
- **Storage pools**
This topic includes overview information about storage pools.

Parent topic: [Concepts for physical storage](#)

Volume groups

Find information about volume groups.

A volume group is a collection of one or more physical volumes of varying sizes and types. A physical volume can belong to only one volume group per system. There can be up to 4096 active volume groups on the Virtual I/O Server.

When a physical volume is assigned to a volume group, the physical blocks of storage media on it are organized into physical partitions of a size determined by the system when you create the volume group. For more information, see Physical Partitions.

When you install the Virtual I/O Server, the root volume group called rootvg is automatically created that contains the base set of logical volumes required to start the system logical partition. The rootvg includes paging space, the journal log, boot data, and dump storage, each in its own separate logical volume. The rootvg has attributes that differ from user-defined volume groups. For example, the rootvg cannot be imported or exported. When using a command or procedure on the rootvg, you must be familiar with its unique characteristics.

Some frequently used volume group commands are as follows:

Command	Description
mkvg	Creates a volume group
extendvg	Adds a physical volume to a volume group
chvg	Changes the attributes of a volume group
reducevg	Removes a physical volume from a volume group
lsvg	Displays information about a volume group
exportvg	Exports the definition of a volume group
importvg	Imports a new volume group definition
syncvg	Synchronizes logical volume copies that are not current
activatevg	Activates a volume group
deactivatevg	Deactivates a volume group

Small systems might require only one volume group to contain all of the physical volumes (beyond the rootvg volume group). You can create separate volume groups to make maintenance easier because groups other than the one being serviced can remain active. Because the rootvg must always be online, it contains only the minimum number of physical volumes necessary for system operation. It is recommended that the rootvg not be used for client data.

You can move data from one physical volume to other physical volumes in the same volume group by using the `migratepv` command. This command allows you to free a physical volume so it can be removed from the volume group. For example, you could move data from a physical volume that is to be replaced. For more information, see the [migratepv](#) command description.

Parent topic: [Logical volumes](#)

Physical partitions

This topic contains information about physical partitions.

When you add a physical volume to a volume group, the physical volume is partitioned into contiguous, equal-sized units of space called *physical partitions*. A physical partition is the smallest unit of storage space allocation and is a contiguous space on a physical volume.

Physical volumes inherit the volume group's physical partition size.

Parent topic: [Logical volumes](#)

Logical partitions

This topic contains information logical storage partitions.

When you create a logical volume, you specify its size in megabytes or gigabytes. The system allocates the number of logical partitions that are required to create a logical volume of at least the specified size. A logical partition is one or two physical partitions, depending on whether the logical volume is defined with mirroring enabled. If mirroring is disabled, there is only one copy of the logical volume (the default). In this case, there is a direct mapping of one logical partition to one physical partition. Each instance, including the first, is called a copy.

Parent topic: [Logical volumes](#)

Quorums

Find information about quorums.

A quorum exists when a majority of Volume Group Descriptor Areas and Volume Group Status Areas (VGDA/VGSA) and their disks are active. A quorum ensures data integrity of the VGDA/VGSA in the event of a disk failure. Each physical disk in a volume group has at least one VGDA/VGSA. When a volume group is created onto a single disk, the volume group initially has two VGDA/VGSA on the disk. If a volume group consists of two disks, one disk still has two VGDA/VGSA, but the other disk has one VGDA/VGSA. When the volume group is made up of three or more disks, each disk is allocated just one VGDA/VGSA.

A quorum is lost when enough disks and their VGDA/VGSA are unreachable so that a 51% majority of VGDA/VGSA no longer exists.

When a quorum is lost, the volume group deactivates itself so that the disks are no longer accessible by the LVM. This prevents further disk I/O to that volume group so that data is not lost or assumed to be written when physical problems occur. As a result of the deactivation, the user is notified in the error log that a hardware error has occurred and service must be performed.

A volume group that has been deactivated because its quorum has been lost can be reactivated by using the `activatevg -f` command.

Parent topic: [Logical volumes](#)

Storage pools

This topic includes overview information about storage pools.

In Virtual I/O Server Version 1.2 and later, storage pools are available. Similar to volume groups, storage pools are collections of one or more physical volumes. The physical volumes that comprise a storage pool can be of varying sizes and types. Using storage pools, you are no longer required to have extensive knowledge on how to manage volume groups and logical volumes to create and assign logical storage to a client partition. Devices created using the storage pool are not limited to the size of the individual physical volumes.

Storage pools are created and managed using the following commands:

Table 1. Storage pool commands and their descriptions

Command	Description
mksp	Creates a storage pool
chsp	Changes the characteristics of a storage pool
lssp	Displays information about a storage pool
mkdbsp	Carves storage out of a pool and assigns it to a virtual SCSI adapter as a backing device
rmdbsp	Disassociates a backing device from its virtual SCSI adapter and removes it from the system

There is a single default storage pool for each Virtual I/O Server partition that can be modified only by the prime administrator. Unless explicitly specified otherwise, the storage pool commands will operate on the default storage pool. This can be useful on systems that contain most or all of its backing devices in a single storage pool.

Note: Storage pools cannot be used when assigning whole physical volumes as backing devices.

Parent topic: [Logical volumes](#)

Optical devices

Optical devices can be exported by the Virtual I/O Server. This topic gives information about what types of optical devices are supported.

The Virtual I/O Server supports exporting optical SCSI devices. These are referred to as a *virtual SCSI optical devices*. Virtual optical devices might be backed by DVD drives. Depending on the backing device, the Virtual I/O Server will export a virtual optical device with one of following profiles:

- DVD-ROM
- DVD-RAM

The virtual optical device can be assigned to only one client logical partition at a time. In order to use the device on a different client logical partition, it must first be removed from its current partition and reassigned to the partition that will use the device. **Parent topic:** [Concepts for physical storage](#)

Concepts for virtual storage

Disks and optical devices are supported as virtual SCSI devices. This topic describes how those devices function in a virtualized environment and provides information on what devices are supported.

- **Disk**
Disk devices can be exported by the Virtual I/O Server. This topic gives information about what types of disks and configurations are supported.
- **Optical**
Optical devices can be exported by the Virtual I/O Server. This topic gives information about what types of optical devices are supported.

Parent topic: [Concepts for Virtual SCSI](#)

Disk

Disk devices can be exported by the Virtual I/O Server. This topic gives information about what types of disks and configurations are supported.

The Virtual I/O Server supports exporting disk SCSI devices. These are referred to as *virtual SCSI disks*. All virtual SCSI disks must be backed by physical storage. Two different types of physical storage can be used to back virtual disks:

- Virtual SCSI disk backed by a physical disk
- Virtual SCSI disk backed by a logical volume

Regardless of whether the virtual SCSI disk is backed by a physical disk or a logical volume, all standard SCSI rules apply to the device. The virtual SCSI device will behave as a standard SCSI-compliant disk device, and it can serve as a boot device or a Network Installation Management (NIM) target, for example.

Virtual SCSI Client Adapter Path Timeout

The Virtual SCSI (VSCSI) Client Adapter Path Timeout feature allows the client adapter to detect if a Virtual I/O Server is not responding to I/O requests. It is recommended that you enable this feature only in configurations in which devices are available to a client partition from multiple Virtual I/O Servers. These configurations could be either configurations where Multipath I/O (MPIO) is being used or where a volume group is being mirrored by devices on multiple Virtual I/O Servers.

If no I/O requests issued to the VSCSI server adapter have been serviced within the number of seconds specified by the VSCSI path timeout value, one more attempt is made to contact the VSCSI server adapter, waiting up to 60 seconds for a response.

If, after 60 seconds, there is still no response from the server adapter, all outstanding I/O requests to that adapter are failed and an error is written to the client partition error log. If MPIO is being used, the MPIO Path Control Module will retry the I/O requests down another path. Otherwise, the failed requests will be returned to the applications. If the devices on this adapter are part of a mirrored volume group, those devices will be marked as *missing* and the Logical Volume Manager logs errors in the client partition error log. If one of the failed devices is the root volume group (rootvg) for the partition, and the rootvg is not available via another path or is not being mirrored on another Virtual I/O Server, the client partition is likely to shut down. The VSCSI client adapter attempts to reestablish communication with the Virtual I/O Server and logs a message in the system error log when it is able to do so. Mirrored volume groups must be manually resynchronized by running the **varyonvg** command when the missing devices are once again available.

A configurable VSCSI client adapter ODM attribute, **vscsi_path_to**, is provided. This attribute is used to both indicate if the feature is enabled and to store the value of the path timeout if the feature is enabled.

The system administrator sets the ODM attribute to 0 to disable the feature, or to the time, in seconds, to wait before checking if the path to the server adapter has failed. If the feature is enabled, a minimum setting of 30 seconds is required. If a setting between 0 and 30 seconds is entered, the value will be changed to 30 seconds upon the next adapter reconfiguration or reboot.

This feature is disabled by default, thus the default value of **vscsi_path_to** is 0. Exercise careful consideration when setting this value, keeping in mind that when the VSCSI server adapter is servicing the I/O request, the storage device the request is being sent to may be either local to the VIO Server or on a SAN.

The **vscsi_path_to** client adapter attribute can be set by using the SMIT utility or by using the **chdev -P** command. The attribute setting can also be viewed by using SMIT or the **lsattr** command. The setting will not

take affect until the adapter is reconfigured or the machine is rebooted.

Parent topic: [Concepts for virtual storage](#)

Optical

Optical devices can be exported by the Virtual I/O Server. This topic gives information about what types of optical devices are supported.

The Virtual I/O Server supports exporting physical optical devices to client partitions. These are referred to as *virtual SCSI optical devices*. Virtual optical devices may be backed by DVD drives. Depending on the backing device, the Virtual I/O Server will export a virtual optical device with one of following profiles:

- DVD-ROM
- DVD-RAM

The virtual optical device can be assigned to only one client logical partition at any given time. To use the device on a different client logical partition, it must first be removed from its current partition and reassigned to the partition that will use the device.

Virtual optical devices will always appear as SCSI devices on the client logical partitions regardless of whether the device type exported from the Virtual I/O Server is a SCSI, IDE, or USB device.

Parent topic: [Concepts for virtual storage](#)

Concepts for mapping devices

Mapping devices are used to facilitate the mapping of physical resources to a virtual device.

Parent topic: [Concepts for Virtual SCSI](#)

Concepts for virtual networking

Use this section to find information about virtual Ethernet, shared Ethernet adapter, shared Ethernet adapter failover, link aggregation, and VLAN.

Virtual Ethernet technology enables IP-based communication between logical partitions on the same system using virtual local area network (VLAN)-capable software switch systems. Shared Ethernet adapter technology enables the logical partitions to communicate with other systems outside the hardware unit without assigning physical Ethernet slots to the logical partitions.

- **[Introduction to virtual networking](#)**
This topic introduces virtual networking concepts. Each concept introduced will be discussed in more detail in subsequent sections.
- **[Virtual local area networks \(VLAN\)](#)**
Virtual local area networks (VLAN) allows the physical network to be logically segmented.
- **[Virtual Ethernet adapters](#)**
Virtual Ethernet adapters allow client logical partitions to send and receive network traffic without having a physical Ethernet adapter.
- **[Shared Ethernet adapters](#)**
Shared Ethernet adapters on the Virtual I/O Server logical partition allow virtual Ethernet adapters on client logical partitions to send and receive outside network traffic.

Parent topic: [Concepts for the Virtual I/O Server](#)

Introduction to virtual networking

This topic introduces virtual networking concepts. Each concept introduced will be discussed in more detail in subsequent sections.

Virtual Ethernet technology allows logical partitions within the same system to communicate without having to use physical Ethernet adapters. Virtual Ethernet adapters are created by using the HMC and configured by using the Virtual I/O Server command-line interface. Virtual Ethernet adapters can be used without using the Virtual I/O Server, but the logical partitions will not be able to communicate with external systems or logical partitions. The Integrated Virtualization Manager can also be used to create and manage virtual Ethernet adapters. For more information, see [Configuring virtual Ethernet bridges on the managed system in the Partitioning with the Integrated Virtualization Manager](#) topic.

Logical partitions that require access to the external network must have either a dedicated Ethernet adapter or a virtual Ethernet adapter that sends and receives packets through a Shared Ethernet adapter on the Virtual I/O Server partition. Multiple client logical partitions can share this resource. The shared Ethernet adapter acts like a layer 2 bridge to forward outbound packets received from a virtual Ethernet adapter to the external network and forward inbound packets to the appropriate client logical partition over the virtual Ethernet link to that partition.

Consider using shared Ethernet on the Virtual I/O Server when the capacity or the bandwidth requirements of the individual partitions is inconsistent or is less than the total bandwidth of a physical Ethernet adapter. Partitions that use the full bandwidth or capacity of a physical Ethernet adapter should use dedicated Ethernet adapters.

The shared Ethernet adapter is configured by using the command-line interface on the Virtual I/O Server. Examples are shown later in this section.

The shared Ethernet adapter can be configured as a link aggregation device and with virtual local area network (VLAN) tagging. Link aggregation helps provide more throughput over a single IP address than would be possible with a single Ethernet adapter. For example, multiple gigabit Ethernet adapters could be configured as a single IP address on the Virtual I/O Server. Link aggregation can also help provide more redundancy because individual links might fail, and the link aggregation device will fail over to another adapter in the device to maintain connectivity.

Parent topic: [Concepts for virtual networking](#)

Virtual local area networks (VLAN)

Virtual local area networks (VLAN) allows the physical network to be logically segmented.

VLAN is a method to logically segment a physical network so that layer 2 connectivity is restricted to members that belong to the same VLAN. This separation is achieved by tagging Ethernet packets with their VLAN membership information and then restricting delivery to members of that VLAN. VLAN is described by the IEEE 802.1Q standard.

The VLAN tag information is referred to as VLAN ID (VID). Ports on a switch are configured as being members of a VLAN designated by the VID for that port. The default VID for a port is referred to as the Port VID (PVID). The VID can be added to an Ethernet packet either by a VLAN-aware host, or by the switch in the case of VLAN-unaware hosts. Ports on an Ethernet switch must therefore be configured with information indicating whether the host connected is VLAN-aware.

For VLAN-unaware hosts, a port is set up as untagged and the switch will tag all packets entering through that port with the Port VLAN ID (PVID). It will also untag all packets exiting that port before delivery to the VLAN unaware host. A port used to connect VLAN-unaware hosts is called an *untagged port*, and it can be a member of only a single VLAN identified by its PVID. Hosts that are VLAN-aware can insert and remove their own tags and can be members of more than one VLAN. These hosts are typically attached to ports that do not remove the tags before delivering the packets to the host, but will insert the PVID tag when an untagged packet enters the port. A port will only allow packets that are untagged or tagged with the tag of one of the VLANs that the port belongs to. These VLAN rules are in addition to the regular media access control (MAC) address-based forwarding rules followed by a switch. Therefore, a packet with a broadcast or multicast destination MAC is also delivered to member ports that belong to the VLAN that is identified by the tags in the packet. This mechanism ensures the logical separation of the physical network based on membership in a VLAN.

Parent topic: [Concepts for virtual networking](#)

Virtual Ethernet adapters

Virtual Ethernet adapters allow client logical partitions to send and receive network traffic without having a physical Ethernet adapter.

Virtual Ethernet adapters are connected to an IEEE 802.1q virtual Ethernet switch. Using this switch function, logical partitions can communicate with each other by using virtual Ethernet adapters and assigning VIDs that enable them to share a common logical network. The virtual Ethernet adapters are created and the VID assignments are done using the Hardware Management Console (HMC). The system transmits packets by copying the packet directly from the memory of the sender partition to the receive buffers of the receiver partition without any intermediate buffering of the packet.

Parent topic: [Concepts for virtual networking](#)

Shared Ethernet adapters

Shared Ethernet adapters on the Virtual I/O Server logical partition allow virtual Ethernet adapters on client logical partitions to send and receive outside network traffic.

While virtual Ethernet technology can provide communication between logical partitions on the same system, network access outside the system requires physical adapters. The Virtual I/O Server can provide network services to logical partitions without requiring each partition to own a physical device. The network access component of the Virtual I/O Server is called the *shared Ethernet adapter*.

A shared Ethernet adapter is a bridge between a physical Ethernet adapter or link aggregation and one or more virtual Ethernet adapters on the Virtual I/O Server. A shared Ethernet adapter enables logical partitions on the virtual Ethernet to share access to the physical Ethernet and communicate with stand-alone servers and logical partitions on other systems. The shared Ethernet adapter provides this access by connecting the internal VLANs with the VLANs on the external switches. This enables logical partitions to share the IP subnet with standalone systems and other external logical partitions. The shared Ethernet adapter processes packets at layer 2, so the original MAC address and VLAN tags of the packet are visible to other systems on the physical network.

Parent topic: [Concepts for virtual networking](#)

Virtual I/O Server management

This topic contains information about Virtual I/O Server management interfaces, such as the Virtual I/O Server command-line interface and the Integrated Virtualization Manager. Virtual I/O Server user types are also explained

- [Integrated Virtualization Manager](#)
Use this topic to find information about the Integrated Virtualization Manager.
- [Virtual I/O Server command-line interface](#)
Use this topic to find information about the Virtual I/O Server command-line interface.
- [User types](#)
Use this topic to provide information about Virtual I/O Server user types and their user permissions.

Parent topic: [Concepts for the Virtual I/O Server](#)

Integrated Virtualization Manager

Use this topic to find information about the Integrated Virtualization Manager.

The Integrated Virtualization Manager provides a graphical user interface available for use in environments where no HMC is present. You can use Integrated Virtualization Manager to perform system management tasks, such as creating partitions and assigning resources to those partitions. Where no HMC is present, the Integrated Virtualization Manager is installed and activated when the Virtual I/O Server is installed. Many of the Virtual I/O Server tasks can be performed using the Integrated Virtualization Manager.

Note: The Virtual I/O Server command-line interface is still accessible to users in a Integrated Virtualization Manager operating environment.

For more information about the Integrated Virtualization Manager, see [Partitioning with Integrated Virtualization Manager](#) and [Managing Integrated Virtualization Manager](#).

Parent topic: [Virtual I/O Server management](#)

Virtual I/O Server command-line interface

Use this topic to find information about the Virtual I/O Server command-line interface.

The Virtual I/O Server is configured and managed through a command-line interface. In environments where no HMC is present, some Virtual I/O Server tasks can also be performed using the Integrated Virtualization Manager. All aspects of Virtual I/O Server administration can be accomplished through the command-line interface, including the following:

- Device management (physical, virtual, LVM)
- Network configuration
- Software installation and update
- Security
- User management
- Maintenance tasks

In addition, in Integrated Virtualization Manager manager environments, the Virtual I/O Server command-line interface is used for partition management.

For a detailed description of each Virtual I/O Server command, see [Virtual I/O Server command descriptions](#).

The first time you log in to the Virtual I/O Server, use the padmin user ID, which is the prime administrator user ID. You will be prompted for a new password.

Restricted shell

Upon logging in, you will be placed into a restricted Korn shell. The restricted Korn shell works in the same way as a standard Korn shell, except that you cannot do the following:

- Change the current working directory
- Set the value of the SHELL, ENV, or PATH variables
- Specify the path name of the command that contains a forward slash (/)
- Redirect output of a command using any of the following characters: >, >|, <>, >>

As a result of these restrictions, you will not be able to execute commands that are not accessible to your PATH variables. In addition, these restrictions prevent you from sending command output directly to a file. Instead, command output can be piped to the [tee](#) command.

After you log in, you can type `help` to get information about the supported commands. For example, to get help on the [errlog](#) command, type `help errlog`.

Execution Mode

The Virtual I/O Server command-line interface functions similarly to a standard command-line interface. Commands are issued with appropriate accompanying flags and parameters. For example, to list all adapters, type the following:

```
lsdev -type adapter
```

In addition, scripts can be run within the Virtual I/O Server command-line interface environment.

Note: For command reference information for the Virtual I/O Server, see [Virtual I/O Server command descriptions](#).

In addition to the Virtual I/O Server command-line interface commands, the following standard shell commands are provided.

Table 1. Standard shell commands and their functions

Command	Function
awk	Matches patterns and performs actions on them.
cat	Concatenates or displays files.
chmod	Changes file modes.
cp	Copies files.
date	Displays the date and time.
grep	Searches a file for a pattern.
ls	Displays the contents of a directory
mkdir	Makes a directory.
man	Displays manual entries for the Virtual I/O Server commands.
more	Displays the contents of files one screen at a time.
rm	Removes files.
sed	Provides a stream editor.
stty	Sets, resets, and reports workstation operating parameters.
tee	Displays the output of a program and copies it to a file.
vi	Edits files with full screen display.
wc	Counts the number of lines, words, and bytes or characters in a file
who	Identifies the users currently logged in.

As each command is executed, the user log and the global command log are updated.

The user log will contain a list of each Virtual I/O Server command, including arguments, that a user has executed. One user log for each user in the system is created. This log is located in the user's home directory and can be viewed by using either the [cat](#) or the [vi](#) commands.

The global command log (GCL) is made up of all the Virtual I/O Server command-line interface commands executed by all users, including arguments, the date and time the command was executed, and from which user ID it was executed. The GCL is viewable only by the padmin user ID, and it can be viewed by using the [lsgcl](#) command. If the global command log exceeds 1 MB, the log will be truncated to 250 KB to prevent the file system from reaching capacity.

Note: Integrated Virtualization Manager commands are audited in a separate place and are viewable either in Application Logs, or by running the following command from the command line: `lssvcevents -t console`

```
--filter severities=audit
```

For command reference information for the Virtual I/O Server, see [Virtual I/O Server command descriptions](#).

Parent topic: [Virtual I/O Server management](#)

User types

Use this topic to provide information about Virtual I/O Server user types and their user permissions.

The Virtual I/O Server has the following user types: prime administrator, system administrator, service representative user, and development engineer user. After installation, the only user type that is active is the prime administrator.

Prime administrator

The prime administrator (padmin) user ID is the only user ID that is enabled after installation of the Virtual I/O Server and can run every Virtual I/O Server command. There can be only one prime administrator in the Virtual I/O Server.

System administrator

The system administrator user ID is created by executing the [mkuser](#) command from the padmin user ID. There is no limit to the number of system administrator IDs that can be created. The system administrator user ID has access to all commands except the following:

- [cleargcl](#)
- [lsfailedlogin](#)
- [lsgcl](#)
- [mirrorios](#)
- [mkuser](#)
- [oem_setup_env](#)
- [rmuser](#)
- [shutdown](#)
- [unmirrorios](#)

Service representative

The SR user is created to allow an service representative (SR) to log in to the system to perform diagnostic routines. SR users are created by running the [mkuser](#) command with the -sr flag.

Upon logging in, the SR user is placed directly into the diagnostic menus.

Development engineer

The DE user is created to allow a development engineer (DE) to log in to the system and debug problems. DE users are created by running the **mkuser** command with the **-de** flag.

Parent topic: [Virtual I/O Server management](#)

Planning for the Virtual I/O Server

Use this topic to help gain an understanding of what to consider when planning for the Virtual I/O Server. In this section, you will find information about planning for the Virtual I/O Server.

- **Specifications**
This topic defines the range of configuration possibilities, including the minimum number of resources needed and the maximum number of resources allowed.
- **Limitations and restrictions**
Find Virtual I/O Server configuration limitations.
- **Capacity planning**
This topic includes capacity-planning considerations for the Virtual I/O Server, including information about hardware resources and limitations.
- **Redundancy considerations**
Redundancy options are available at several levels in the virtual I/O environment. Multipathing and RAID redundancy options exist for both the Virtual I/O Server and client partitions. Ethernet link aggregation is also an option for the client partitions, and the Virtual I/O Server provides shared Ethernet adapter failover. There is also support for node failover (HACMP) for nodes using virtual I/O resources.
- **Security considerations**
Review the security considerations for Virtual SCSI, virtual Ethernet, and Shared Ethernet adapter and the additional security options available.

Parent topic: [Using the Virtual I/O Server](#)

Specifications

This topic defines the range of configuration possibilities, including the minimum number of resources needed and the maximum number of resources allowed.

To activate the Virtual I/O Server, the Advanced POWER Virtualization hardware feature is required. A logical partition with enough resources to share with other partitions is required. The following is a list of minimum hardware requirements that must be available to create the Virtual I/O Server:

Table 1. Resources that are required

Resource	Requirement
Hardware Management Console or Integrated Virtualization Manager	Either the HMC or Integrated Virtualization Manager is required to create the partition and assign resources.
Storage adapter	The server partition needs at least one storage adapter.
Physical disk	The disk must be at least 16 GB. This disk can be shared.
Ethernet adapter	If you want to route network traffic from virtual Ethernet adapters to a shared Ethernet adapter, you need an Ethernet adapter.
Memory	At least 512 MB of memory is required.
Processor	At least .1 processor is required.

The following table defines the limitations for storage management.

Table 2. Limitations for storage management

Category	Limit
Volume groups	4096 per system
Physical volumes	1024 per volume group
Physical partitions	1024 per volume group
Logical volumes	1024 per volume group
Logical partitions	No limit

Parent topic: [Planning for the Virtual I/O Server](#)

Limitations and restrictions

Find Virtual I/O Server configuration limitations.

Logical volumes exported as Virtual SCSI disks are created by using the Virtual I/O Server command-line interface. If a logical volume is exported as a virtual device, all physical volumes that make up the volume group in which the logical volume is contained must be attached to the same adapter. You can ensure this situation by creating volume groups with no more than one physical disk.

Consider the following when implementing Virtual SCSI:

- Virtual SCSI supports Fibre Channel, parallel SCSI, and SCSI RAID devices as backing devices.
- Virtual SCSI does not have any limitations in terms of the number of supported adapters. A maximum of 256 virtual slots can be assigned to a single partition. Every virtual slot that is created requires resources in order to be instantiated. Therefore, the size of the Virtual I/O Server places a limit on the number of virtual adapters that can be configured.
- The SCSI protocol defines mandatory and optional commands. While virtual SCSI supports all of the mandatory commands, not all of the optional commands are supported.
- There are performance implications when using Virtual SCSI devices. Because of the overhead associated with the client/server model, Virtual SCSI uses additional processor cycles when processing I/O requests.
- The Virtual I/O Server is a dedicated partition, to be used only for Virtual I/O Server operations. Other applications cannot run in the Virtual I/O Server partition.
- If there is a resource shortage, performance degradation might occur. If a Virtual I/O Server is serving many resources to other partitions, ensure that enough processor power is available. In case of high workload across virtual Ethernet adapters and virtual disks, partitions might experience delays in accessing resources.
- Logical volumes exported as Virtual SCSI disks are always configured as single path devices on the client partition.
- Logical volumes exported as Virtual SCSI disks that are part of the root volume group (rootvg) are not persistent when the Virtual I/O Server is updated for maintenance. Therefore, before performing an update procedure, ensure that the corresponding clients' virtual disks are backed up. When exporting logical volumes, it is best to export logical volumes from a volume group other than the root volume group.

Consider the following when implementing shared Ethernet adapters:

- Only Ethernet adapters can be shared. Other types of network adapters cannot be shared.
- IP forwarding is not supported on the Virtual I/O Server.

The Virtual I/O Server supports client partitions running only the following operating systems:

- AIX 5.3
- SUSE LINUX Enterprise Server 9 for POWER
- Red Hat Enterprise Linux AS for POWER Version 3
- Red Hat Enterprise Linux AS for POWER Version 4

Parent topic: [Planning for the Virtual I/O Server](#)

Capacity planning

This topic includes capacity-planning considerations for the Virtual I/O Server, including information about hardware resources and limitations.

Client partitions might use virtual devices, dedicated devices, or a combination of both. Before you begin to configure and install the Virtual I/O Server and client partitions, plan what resources each partition will use. Throughput requirements and overall workload must be considered when deciding whether to use virtual or dedicated devices and when allocating resources to the Virtual I/O Server. Compared to dedicated SCSI disks, Virtual SCSI disks might achieve similar throughput numbers depending on several factors, including workload and Virtual I/O Server resources. However, Virtual SCSI devices generally have higher processor utilization when compared with directly attached storage.

- **Planning for shared Ethernet adapters**
Use this section to find capacity-planning and performance information for Shared Ethernet adapter. This section contains planning information and performance considerations for using shared Ethernet adapters on the Virtual I/O Server.
- **Planning for Virtual SCSI**
Find capacity-planning and performance information for Virtual SCSI.

Parent topic: [Planning for the Virtual I/O Server](#)

Planning for shared Ethernet adapters

Use this section to find capacity-planning and performance information for Shared Ethernet adapter. This section contains planning information and performance considerations for using shared Ethernet adapters on the Virtual I/O Server.

- **Network requirements**
This topic includes information you need in order to accurately size your shared Ethernet adapter environment.
- **Shared Ethernet adapter sizing considerations**
Use this section to find information about Ethernet adapter, processor, and memory-sizing considerations.

Parent topic: [Capacity planning](#)

Network requirements

This topic includes information you need in order to accurately size your shared Ethernet adapter environment.

To plan for using Shared Ethernet adapters, you must determine your network needs. This section gives overview information of what should be considered when sizing the shared Ethernet adapter environment. Sizing the Virtual I/O Server for the shared Ethernet adapter involves the following factors:

- Defining the target bandwidth (MB per second), or transaction rate requirements (operations per second). The target performance of the configuration must be determined from your workload requirements.
- Defining the type of workload (streaming or transaction oriented).
- Identifying the maximum transmission unit (MTU) size that will be used (1500 or jumbo frames).
- Determining if the shared Ethernet adapter will run in a threaded or nonthreaded environment.
- Knowing the throughput rates that various Ethernet adapters can provide (see [Adapter selection](#)).
- Knowing the processor cycles required per byte of throughput or per transaction (see [Processor allocation](#)).

Bandwidth requirement

The primary consideration is determining the target bandwidth on the physical Ethernet adapter of the Virtual I/O Server. This will determine the rate that data can be transferred between the Virtual I/O Server and the client logical partitions. After the target rate is known, the correct type and number of network adapters can be selected. For example, Ethernet adapters of various speeds could be used. One or more adapters could be used on individual networks, or they could be combined using link aggregation.

Workload type

The type of workload to be performed must be considered, whether it is streaming of data for workloads such as file transfer, data backup, or small transaction workloads, such as remote procedure calls. The streaming workload consists of large, full-sized network packets and associated small, TCP acknowledgment packets. Transaction workloads typically involve smaller packets or might involve small requests, such as a URL, and a larger response, such as a Web page. A Virtual I/O Server will need to frequently support streaming and small packet I/O during various periods of time. In that case, approach the sizing from both models.

MTU size

The MTU size of the network adapters must also be considered. The standard Ethernet MTU is 1500 bytes. Gigabit Ethernet and 10 gigabit Ethernet can support 9000-byte MTU jumbo frames. Jumbo frames might reduce the processor cycles for the streaming types of workloads. However, for small workloads, the larger MTU size might not help reduce processor cycles.

Threaded or nonthreaded environment

Use threaded mode when Virtual SCSI will be run on the same Virtual I/O Server partition as Shared Ethernet adapter. Threaded mode helps ensure that Virtual SCSI and the Shared Ethernet adapter can share the processor resource appropriately. However, threading increases instruction-path length, which uses additional processor cycles. If the Virtual I/O Server partition will be dedicated to running shared Ethernet devices (and associated virtual Ethernet devices) only, the adapters should be configured with threading disabled. For more information, see [Processor allocation](#).

Adapter throughput

Knowing the throughput capability of different Ethernet adapters can help you determine which adapters to use as shared Ethernet adapters and how many adapters to use. For more information, see [Adapter selection](#).

Processor entitlement

You must determine how much processor power is required to move data through the adapters at the desired rate. Networking device drivers are typically processor-intensive. Small packets can come in at a faster rate and use more processor cycles than larger packet workloads. Larger packet workloads are typically limited by network wire bandwidth and come in at a slower rate, thus requiring less processor power than small packet workloads for the amount of data transferred.

Parent topic: [Planning for shared Ethernet adapters](#)

Shared Ethernet adapter sizing considerations

Use this section to find information about Ethernet adapter, processor, and memory-sizing considerations.

- **Adapter selection**
Use this section to find the attributes and performance characteristics of various types of Ethernet adapters to help you select which adapters to use in your environment.
- **Processor allocation**
This section contains processor-allocation guidelines for both dedicated processor partitions and shared processor partitions.
- **Memory allocation**
Find information about memory allocation and sizing.

Parent topic: [Planning for shared Ethernet adapters](#)

Adapter selection

Use this section to find the attributes and performance characteristics of various types of Ethernet adapters to help you select which adapters to use in your environment.

This section provides approximate throughput rates for various Ethernet adapters set at various MTU sizes. Use this information to determine which adapters will be needed to configure a Virtual I/O Server. To make this determination, you must know the desired throughput rate of the client logical partitions.

Following are general guidelines for network throughput. These numbers are not specific, but they can serve as a general guideline for sizing. In the following tables, the 100 Mb and 1 Gb speeds are rounded down for estimating.

Table 1. Simplex (one direction) streaming rates

Adapter speed	Approximate throughput rate
10 Mb Ethernet	1 MB/second
100 Mb Ethernet	10 MB/second
1000 Mb Ethernet (Gb Ethernet)	100 MB/second

Table 2. Full duplex (two direction) streaming rates on full duplex network

Adapter speed	Approximate throughput rate
10 Mb Ethernet	2 MB/second
100 Mb Ethernet	20 MB/second
1000 Mb Ethernet (Gb Ethernet)	150 MB/second

Note: For placement rules and limitations, see [PCI adapter placement for servers system units and expansion units](#).

The following tables list maximum network payload speeds, which are user payload data rates that can be obtained by sockets-based programs for applications that are streaming data. The rates are a result of the network bit rate, MTU size, physical level overhead, data link headers, and TCP/IP headers. A gigahertz-speed processor is assumed. These numbers are optimal for a single LAN. If your network traffic is going through additional network devices, your results might vary.

In the following tables, raw bit rate is the physical media bit rate and does not reflect physical media overheads, such as inter-frame gaps, preamble bits, cell overhead, data link headers, and trailers. These overheads can all reduce the effective usable bit rate of the wire.

Single direction (simplex) TCP streaming rates are rates that can be achieved by sending data from one machine to another in a memory-to-memory test. Full-duplex media can usually perform slightly better than half-duplex media because the TCP acknowledgment packets can flow without contending for the same wire that the data packets are flowing on.

Table 3. Single direction (simplex) TCP streaming rates

Network type	Raw bit rate (Mb)	Payload rate (Mb)	Payload rate (MB)
10 Mb Ethernet, Half Duplex	10	6	0.7
10 Mb Ethernet, Full Duplex	10 (20 Mb full duplex)	9.48	1.13
100 Mb Ethernet, Half Duplex	100	62	7.3
100 Mb Ethernet, Full Duplex	100 (200 Mb full duplex)	94.8	11.3
1000 Mb Ethernet, Full Duplex, MTU 1500	1000 (2000 Mb full duplex)	948	113
1000 Mb Ethernet, Full Duplex, MTU 9000	1000 (2000 Mb full duplex)	989	117.9

Full-duplex TCP streaming workloads have data streaming in both directions. Workloads that can send and receive packets concurrently can take advantage of full duplex media. Some media, for example Ethernet in half-duplex mode, cannot send and receive concurrently, thus they will not perform any better, and can usually degrade performance, when running duplex workloads. Duplex workloads will not increase at a full doubling of the rate of a simplex workload because the TCP acknowledgment packets returning from the receiver must now compete with data packets flowing in the same direction.

Table 4. Two direction (duplex) TCP streaming rates

Network type	Raw bit rate (Mb)	Payload rate (Mb)	Payload rate (MB)
10 Mb Ethernet, Half Duplex	10	5.8	0.7
10 Mb Ethernet, Full Duplex	10 (20 Mb full duplex)	18	2.2
100 Mb Ethernet, Half Duplex	100	58	7
100 Mb Ethernet, Full Duplex	100 (200 Mb full duplex)	177	21.1
1000 Mb Ethernet, Full Duplex, MTU 1500	1000 (2000 Mb full duplex)	1470 (1660 peak)	175 (198 peak)
1000 Mb Ethernet, Full Duplex, MTU 9000	1000 (2000 Mb full duplex)	1680 (1938 peak)	200 (231 peak)

Note:

1. Peak numbers represent optimal throughput with multiple TCP sessions running in each direction.

Other rates are for a single TCP session.

2. 1000 Mb Ethernet (gigabit Ethernet) duplex rates are for the PCI-X adapter in PCI-X slots.
3. Data rates are for TCP/IP using the IPv4 protocol. Adapters with MTU set to 9000 have RFC 1323 enabled.

Parent topic: [Shared Ethernet adapter sizing considerations](#)

Processor allocation

This section contains processor-allocation guidelines for both dedicated processor partitions and shared processor partitions.

Because Ethernet running MTU size of 1500 bytes consumes more processor cycles than Ethernet running Jumbo frames (MTU 9000), the guidelines are different for each situation. In general, the processor utilization for large packet workloads on jumbo frames is approximately half that required for MTU 1500.

If MTU is set to 1500, provide one processor (1.65 Ghz) per Gigabit Ethernet adapter to help reach maximum bandwidth. This equals ten 100-Mb Ethernet adapters if you are using smaller networks. For smaller transaction workloads, plan to use one full processor to drive the Gigabit Ethernet workload to maximum throughput. For example, if two Gigabit Ethernet adapters will be used, allocate up to two processors to the partition.

If MTU is set to 9000 (jumbo frames), provide 50% of one processor (1.65 Ghz) per Gigabit Ethernet adapter to reach maximum bandwidth. Small packet workloads should plan to use one full processor to drive the Gigabit Ethernet workload. Jumbo frames have no effect on the small packet workload case.

Shared Ethernet adapter using a dedicated processor partition

The sizing provided is divided into two workload types: TCP streaming and TCP request and response. Both MTU 1500 and MTU 9000 networks were used in the sizing, which is provided in terms of machine cycles per byte of throughput for streaming or per transaction for request/response workloads.

The data in the following tables was derived using the following formula:

$$(\text{number of processors} \times \text{processor_utilization} \times \text{processor clock frequency}) / \text{Throughput rate in bytes per second or transaction per second} = \text{cycles per Byte or transaction.}$$

For the purposes of this test, the numbers were measured on a logical partition with one 1.65 Ghz processor with simultaneous multi-threading (SMT) enabled.

For other processor frequencies, the numbers in these tables can be scaled by the ratio of the processor frequencies for approximate values to be used for sizing. For example, for a 1.5 Ghz processor speed, use $1.65/1.5 \times$ cycles per byte value from the table. This example would result in a value of 1.1 times the value in the table, thus requiring 10% more cycles to adjust for the 10% slower clock rate of the 1.5 Ghz processor.

To use these values, multiply your required throughput rate (in bytes or transactions) by the cycles per byte value in the following tables. This result will give you the required machine cycles for the workload for a 1.65 Ghz speed. Then adjust this value by the ratio of the actual machine speed to this 1.65 Ghz speed. To find the number of processors, divide the result by 1,650,000,000 cycles (or the cycles rate if you adjusted to a different speed machine). You would need the resulting number of processors to drive the workload.

For example, if the Virtual I/O Server must deliver 200 MB of streaming throughput, the following formula would be used:

$$200 \times 1024 \times 1024 \times 11.2 = 2,348,810,240 \text{ cycles} / 1,650,000,000 \text{ cycles per processor} = 1.42 \text{ processors.}$$

In round numbers, it would require 1.5 processors in the Virtual I/O Server to handle this workload. Such a workload could then be handled with either a 2-processor dedicated partition or a 1.5-processor shared-processor partition.

The following tables show the machine cycles per byte for a TCP-streaming workload.

Table 1. Shared Ethernet with threading option enabled

Type of Streaming	MTU 1500 rate and processor utilization	MTU 1500, cycles per byte	MTU 9000 rate and processor utilization	MTU 9000, cycles per byte
Simplex	112.8 MB at 80.6% processor	11.2	117.8 MB at 37.7% processor	5
Duplex	162.2 MB at 88.8% processor	8.6	217 MB at 52.5% processor	3.8

Table 2. Shared Ethernet with threading option disabled

Type of Streaming	MTU 1500 rate and processor utilization	MTU 1500, cycles per byte	MTU 9000 rate and processor utilization	MTU 9000, cycles per byte
Simplex	112.8 MB at 66.4% processor	9.3	117.8 MB at 26.7% processor	3.6
Duplex	161.6 MB at 76.4% processor	7.4	216.8 MB at 39.6% processor	2.9

The following tables show the machine cycles per transaction for a request and response workload. A transaction is defined as a round-trip request and reply size.

Table 3. Shared Ethernet with threading option enabled

Size of transaction	Transactions per second and Virtual I/O Server utilization	MTU 1500 or 9000, cycles per transaction
Small packets (64 bytes)	59,722 TPS at 83.4% processor	23,022
Large packets (1024 bytes)	51,956 TPS at 80% processor	25,406

Table 4. Shared Ethernet with threading option disabled

Size of transaction	Transactions per second and Virtual I/O Server utilization	MTU 1500 or 9000, cycles per transaction
Small packets (64 bytes)	60,249 TPS at 65.6% processor	17,956
Large packets (1024 bytes)	53,104 TPS at 65% processor	20,196

The preceding tables demonstrate that the threading option of the shared Ethernet adds overhead. It is approximately 16% to 20% more overhead for MTU 1500 streaming and 31% to 38% more overhead for MTU 9000. The threading option has more overhead at lower workloads due to the threads being started for each packet. At higher workload rates, like full duplex or the request and response workloads, the threads can run longer without waiting and being redispached. The thread option is a per-shared Ethernet option that can be configured by Virtual I/O Server commands. Disable the thread option if the shared Ethernet is running in a Virtual I/O Server partition by itself (without Virtual SCSI in the same partition).

You can enable or disable threading using the `-attr thread` option of the `mkvdev` command. To enable threading, use the `-attr thread=1` option. To disable threading, use the `-attr thread=0` option. For example, the following command disables threading for shared Ethernet adapter `ent1`:

```
mkvdev -sea ent1 -vadapter ent5 -default ent5 -defaultid 1 -attr thread=0
```

Sizing a Virtual I/O Server for shared Ethernet on a shared processor partition

Creating a shared-processor partition for a Virtual I/O Server can be done if the Virtual I/O Server is running slower-speed networks (for example 10/100 Mb) and a full processor partition is not needed. It is recommended that this be done only if the Virtual I/O Server workload is less than half a processor or if the workload is inconsistent. Configuring the Virtual I/O Server partition as uncapped might also allow it to use more processor cycles as needed to handle inconsistent throughput. For example, if the network is used only when other processors are idle, the Virtual I/O Server partition might be able to use other machine cycles and could be created with minimal processor to handle light workload during the day but the uncapped processor could use more machine cycles at night.

If you are creating a Virtual I/O Server in a shared-processor partition, add additional processor entitlement as a sizing contingency.

Parent topic: [Shared Ethernet adapter sizing considerations](#)

Memory allocation

Find information about memory allocation and sizing.

In general, 512 MB of memory per partition is sufficient for most configurations. Enough memory must be allocated for the Virtual I/O Server data structures. Ethernet adapters and virtual devices use dedicated receive buffers. These buffers are used to store the incoming packets, which are then sent over the outgoing device.

A physical Ethernet adapter typically uses 4 MB for MTU 1500 or 16 MB for MTU 9000 for dedicated receive buffers for gigabit Ethernet. Other Ethernet adapters are similar. Virtual Ethernet, typically uses 6 MB for dedicated receive buffers. However, this number can vary based on workload. Each instance of a physical or virtual Ethernet would need memory for this number of buffers. In addition, the system has an mbuf buffer pool per processor that is used if additional buffers are needed. These mbufs typically occupy 40 MB.

Parent topic: [Shared Ethernet adapter sizing considerations](#)

Planning for Virtual SCSI

Find capacity-planning and performance information for Virtual SCSI.

Different I/O subsystems have different performance qualities, as does Virtual SCSI. This section discusses the performance differences between physical and virtual I/O. The following topics are described in this section:

- [Virtual SCSI latency](#)
Find information about Virtual SCSI latency.
- [Virtual SCSI bandwidth](#)
View information about Virtual SCSI bandwidth.
- [Virtual SCSI sizing considerations](#)
Understand the processor and memory-sizing considerations when implementing Virtual SCSI .

Parent topic: [Capacity planning](#)

Virtual SCSI latency

Find information about Virtual SCSI latency.

I/O latency is the amount of time that passes between the initiation and completion of a disk I/O operation. For example, consider a program that performs 1000 random disk I/O operations, one at a time. If the time to complete an average operation is 6 milliseconds, the program runs in no fewer than 6 seconds. However, if the average response time is reduced to 3 milliseconds, the run time might be reduced by 3 seconds. Applications that are multithreaded or use asynchronous I/O might be less sensitive to latency, but in most circumstances, lower latency can help improve performance.

Because Virtual SCSI is implemented as a client and server model, there is some latency overhead that does not exist with directly attached storage. The overhead might range from 0.03 to 0.06 milliseconds per I/O operation depending primarily on the block size of the request. The average latency overhead is comparable for both physical disk and logical volume-backed virtual drives. The latency experienced when using a Virtual I/O Server in a shared-processor partition can be higher and more variable than using a Virtual I/O Server in a dedicated partition. For additional information about the performance differences between dedicated partitions and shared-processor partitions, see [Virtual SCSI sizing considerations](#).

The following table identifies latency overheads for different block-size transmissions on both physical disk and logical-volume-backed Virtual SCSI disks.

Table 1. Increase in disk I/O response time based on block size (in milliseconds)

Backing type	4 K	8 K	32 K	64 K	128 K
Physical disk	0.032	0.033	0.033	0.040	0.061
Logical volume	0.035	0.036	0.034	0.040	0.063

The average disk-response time increases as the block size increases. The latency increases for a Virtual SCSI operation are relatively greater on smaller block sizes because of their shorter response time.

Parent topic: [Planning for Virtual SCSI](#)

Virtual SCSI bandwidth

View information about Virtual SCSI bandwidth.

I/O bandwidth is the maximum amount of data that can be read or written to a storage device in a unit of time. Bandwidth can be measured from a single thread or from a set of threads running concurrently. Although many customer applications are more sensitive to latency than bandwidth, bandwidth is crucial for many typical operations, such as backing up and restoring persistent data.

The following table compares the results of bandwidth tests for Virtual SCSI and physical I/O performance. In the tests, a single thread operates sequentially on a constant file that is 256 MB in size with a Virtual I/O Server running in a dedicated partition. More I/O operations are issued when reading or writing to the file using a small block size as compared to a larger block size. The test was conducted using a storage server with feature code 6239 (type 5704/0625) and a 2-gigabit Fibre Channel adapter attached to one RAID0 LUN that is composed of 5 physical disks from a DS4400 disk system (formerly a FAStT700). The table shows the comparison of measured bandwidth in megabytes per second (MB/s) using Virtual SCSI and local attachment for reads with varying block sizes of operations. The difference between virtual I/O and physical I/O in these tests is attributable to the increased latency when using virtual I/O. Because of the larger number of operations, the bandwidth measured with small block sizes is lower than with large block sizes.

Table 1. Physical and Virtual SCSI bandwidth comparison (in MB/s)

I/O type	4 K	8 K	32 K	64 K	128 K
Virtual	20.3	35.4	82.6	106.8	124.5

Physical	24.3	41.7	90.6	114.6	132.6
----------	------	------	------	-------	-------

Parent topic: [Planning for Virtual SCSI](#)

Virtual SCSI sizing considerations

Understand the processor and memory-sizing considerations when implementing Virtual SCSI .

When you are designing and implementing a Virtual SCSI application environment, consider the following sizing issues:

- The amount of memory allocated to the Virtual I/O Server
- The processor entitlement of the Virtual I/O Server
- Whether the Virtual I/O Server is run as a shared-processor partition or as a dedicated processor partition

The processor impacts of using virtual I/O on the client are insignificant. The processor cycles run on the client to perform a Virtual SCSI I/O operation are comparable to that of a locally attached I/O device. Thus, there is no increase or decrease in sizing on the client partition for a known task. These sizing techniques do not anticipate combining the function of shared Ethernet with the Virtual SCSI server. If the two are combined, consider adding resources to account for the shared Ethernet activity with Virtual SCSI .

Virtual SCSI sizing using dedicated processor partitions

The amount of processor entitlement required for a Virtual SCSI server is based on the maximum I/O rates required of it. Because Virtual SCSI servers do not normally run at maximum I/O rates all of the time, the use of surplus processor time is potentially wasted when using dedicated processor partitions. In the first of the following sizing methodologies, you need a good understanding of the I/O rates and I/O sizes required of the Virtual SCSI server. In the second, we will size the Virtual SCSI server based on the I/O configuration.

The sizing methodology used is based on the observation that the processor time required to perform an I/O operating on the Virtual SCSI server is fairly constant for a given I/O size. It is a simplification to make this statement, because different device drivers have subtly varying efficiencies. However, under most circumstances, the I/O devices supported by the Virtual SCSI server are sufficiently similar. The following table shows approximate cycles per second for both physical disk and logical volume operations on a 1.65 Ghz processor. These numbers are measured at the physical processor; simultaneous multi-threading (SMT) operation is assumed. For other frequencies, scaling by the ratio of the frequencies (for example, 1.5 Ghz = 1.65 Ghz / 1.5 Ghz $\hat{=}$ cycles per operation) is sufficiently accurate to produce a reasonable sizing.

Table 1. Approximate cycles per second on a 1.65 Ghz partition

Disk type	4 KB	8 KB	32 KB	64 KB	128 KB
Physical disk	45,000	47,000	58,000	81,000	120,000
Logical volume	49,000	51,000	59,000	74,000	105,000

Consider a Virtual I/O Server that uses three client partitions on physical disk-backed storage. The first client partition requires a maximum of 7,000 8-KB operations per second. The second client partition requires a maximum of 10,000 8-KB operations per second. The third client partition requires a maximum of 5,000 128-KB operations per second. The number of 1.65 Ghz processors for this requirement is approximately $((7,000 \hat{+} 47,000 + 10,000 \hat{+} 47,000 + 5,000 \hat{+} 120,000) / 1,650,000,000) = 0.85$ processors, which rounds up to a single processor when using a dedicated processor partition.

If the I/O rates of the client partitions are not known, you can size the Virtual I/O Server to the maximum I/O rate of the storage subsystem attached. The sizing could be biased toward small I/O operations or large I/O operations. Sizing to maximum capacity for large I/O operations will balance the processor capacity of the Virtual I/O Server to the potential I/O bandwidth of the attached I/O. The negative aspect of this sizing

methodology is that, in nearly every case, more processor entitlement will be assigned to the Virtual I/O Server than it will typically consume.

Consider a case in which a Virtual I/O Server manages 32 physical SCSI disks. An upper limit of processors required can be established based on assumptions about the I/O rates that the disks can achieve. If it is known that the workload is dominated by 8096-byte operations that are random, then assume that each disk is capable of approximately 200 disk I/O operations per second (15k rpm drives). At peak, the Virtual I/O Server would need to serve approximately 32 disks \times 200 I/O operations per second \times 120,000 cycles per operation, resulting in a requirement for approximately 0.19 processor's performance. Viewed another way, a Virtual I/O Server running on a single processor should be capable of supporting more than 150 disks doing 8096-byte random I/O operations.

Alternatively, if the Virtual I/O Server is sized for maximum bandwidth, the calculation results in a higher processor requirement. The difference is that maximum bandwidth assumes sequential I/O. Because disks are more efficient when they are performing large, sequential I/O operations than they are when performing small, random I/O operations, a higher number of I/O operations per second can be performed. Assume that the disks are capable of 50 MB per second when doing 128 kb I/O operations. That situation implies each disk could average 390 disk I/O operations per second. Thus, the amount of processing power necessary to support 32 disks, each doing 390 I/O operations per second with an operation cost of 120,000 cycles ($32 \times 390 \times 120,000 / 1,650,000,000$) results in approximately 0.91 processors. Consequently, a Virtual I/O Server running on a single processor should be capable of driving approximately 32 fast disks to maximum throughput.

Virtual SCSI server sizing using shared processor partitions

Defining Virtual SCSI servers in shared processor partitions allows more specific processor resource sizing and potential recovery of unused processor time by uncapped partitions. However, using shared-processor partitions for Virtual SCSI servers can frequently increase I/O response time and make for somewhat more complex processor entitlement sizings.

The sizing methodology should be based on the same operation costs for dedicated partition I/O servers, with added entitlement for running in shared-processor partitions. Configure the Virtual I/O Server as uncapped, so that, if the Virtual I/O Server is undersized, there is opportunity to get more processor time to serve I/O operations.

Because I/O latency with Virtual SCSI can vary due to a number of conditions, consider the following if a partition has high I/O requirements:

- Configure the partition with physical I/O if the configuration allows.
- If physical I/O is not possible and the system contains enough processors, consider using Virtual SCSI with the Virtual I/O Server in a dedicated processor partition.
- If it is necessary to use a Virtual I/O Server in a shared-processor partition, use as few virtual processors as possible to minimize the overhead of the firmware.

Virtual SCSI server memory sizing

Memory sizing in Virtual SCSI is simplified because there is no caching of file data in the memory of the Virtual SCSI server. Because there is no data caching, the memory requirements for the Virtual SCSI server are fairly modest. With large I/O configurations and very high data rates, a 1 GB memory allocation for the Virtual SCSI server is likely to be sufficient. For low I/O rate situations with a small number of attached disks, 512 MB will most likely suffice.

Parent topic: [Planning for Virtual SCSI](#)

Redundancy considerations

Redundancy options are available at several levels in the virtual I/O environment. Multipathing and RAID redundancy options exist for both the Virtual I/O Server and client partitions. Ethernet link aggregation is also an option for the client partitions, and the Virtual I/O Server provides shared Ethernet adapter failover. There is also support for node failover (HACMP) for nodes using virtual I/O resources.

This section contains information about redundancy for both the client partitions and the Virtual I/O Server. While these configurations help protect from the failure of one of the physical components, such as a disk or network adapter, they might cause the client partition to lose access to its devices if the Virtual I/O Server fails. The Virtual I/O Server can be made redundant by running a second instance of it in another partition. When running two instances of the Virtual I/O Server, you can use LVM mirroring, multipath I/O, network interface backup, or multipath routing with dead gateway detection in the client partition to provide highly available access to virtual resources hosted in separate Virtual I/O Server partitions.

- **Client logical partitions**

This topic includes redundancy considerations for client logical partitions. MPIO, HACMP, and mirroring for the client logical partition are discussed.

- **Virtual I/O Server partition**

Redundancy options for the Virtual I/O Server include multi-pathing, Redundant Array of Independent Disks (RAID) configurations, and link aggregation.

Parent topic: [Planning for the Virtual I/O Server](#)

Client logical partitions

This topic includes redundancy considerations for client logical partitions. MPIO, HACMP, and mirroring for the client logical partition are discussed.

- **Multipath I/O**

View Multipath I/O (MPIO) information for client logical partitions.

- **Mirroring for client logical partitions**

Achieve mirroring for client logical partitions by using two virtual SCSI adapters.

- **High Availability Cluster Multi-Processing**

Learn about High Availability Cluster Multi-Processing (HACMP) in the Virtual I/O Server.

- **Link aggregation devices**

A link aggregation device is a group of network adapters that are grouped to provide redundancy.

- **Shared Ethernet Adapter failover**

Shared Ethernet Adapter failover provides redundancy by configuring a backup Shared Ethernet Adapter on a different Virtual I/O Server partition that can be used if the primary Shared Ethernet Adapter fails. The network connectivity in the client logical partitions continues without disruption.

Parent topic: [Redundancy considerations](#)

Multipath I/O

View Multipath I/O (MPIO) information for client logical partitions.

Multiple Virtual SCSI client adapters in a client logical partition can access the same disk through multiple Virtual I/O Server partitions. This section describes a Virtual SCSI multipath device configuration. If correctly configured, the client recognizes the disk as a multipath device.

Not all Virtual SCSI devices are capable of MPIO. To create an MPIO configuration, the exported device at the Virtual I/O Server must conform to the following rules:

- The device must be backed by a physical volume. Logical volume-backed Virtual SCSI devices are not supported in an MPIO configuration.
- The device must be accessible from multiple Virtual I/O Server partitions.
- The device must be an MPIO-capable device.

Note: MPIO-capable devices are those that contain a unique identifier (UDID) or IEEE volume identifier. For instructions about how to determine whether disks have a UDID or IEEE volume identifier, see [Listing disks with a UDID](#) or [Listing disks with an IEEE volume attribute](#).

When setting up a Virtual SCSI device MPIO configuration on the client logical partition, you must consider the reservation policy of the device on the Virtual I/O Server. To enable an MPIO configuration at the client, none of the Virtual SCSI devices on the Virtual I/O Server should be reserving the SCSI reserve. Ensure the `reserve_policy` attribute of the device is set to `no_reserve`. To determine the reserve policy of a device, type the following command:

```
lsdev -dev diskdevicename -attr reserve_policy
```

If the `reserve_policy` value is anything other than `no_reserve`, it must be changed so that you can use the device in an MPIO configuration on the client logical partition. To set the attribute, use the following command:

```
chdev -dev diskdevicename -attr reserve_policy=no_reserve
```

Failover is the default behavior for MPIO Virtual SCSI disks on the client logical partition. Configuring multipath Virtual SCSI devices on the client logical partition protects the partition against failure of one of the following:

For information about configuring MPIO, see [Scenario: Configuring Multi-Path I/O for AIX client logical partitions](#).

Parent topic: [Client logical partitions](#)

Mirroring for client logical partitions

Achieve mirroring for client logical partitions by using two virtual SCSI adapters.

The client partition can mirror its logical volumes using two virtual SCSI client adapters. Each of these adapters should be assigned to separate Virtual I/O Server partitions. The two physical disks are each attached to a separate Virtual I/O Server partition and made available to the client partition through a Virtual SCSI server adapter. This configuration protects virtual disks in a client partition against the failure of any of the following:

- One physical disk
- One physical adapter
- One Virtual I/O Server

The performance of your system might be impacted when using a RAID 1 configuration.

Parent topic: [Client logical partitions](#)

High Availability Cluster Multi-Processing

Learn about High Availability Cluster Multi-Processing (HACMP) in the Virtual I/O Server.

HACMP and virtual SCSI

Be aware of the following considerations when implementing HACMP and virtual SCSI:

- The volume group must be defined as Enhanced Concurrent Mode. Enhanced Concurrent Mode is the preferred mode for sharing volume groups in HACMP clusters because volumes are accessible by multiple HACMP nodes. If file systems are used on the standby nodes, those file systems are not mounted until the point of failover. If shared volumes are accessed directly (without file systems) in Enhanced Concurrent Mode, these volumes are accessible from multiple nodes, and as a result, access must be controlled at a higher layer.

- If any one cluster node accesses shared volumes through virtual SCSI, then all nodes must. This means that disks cannot be shared between a logical partitions using virtual SCSI and a node directly accessing those disks.
- All volume group configuration and maintenance on these shared disks is done from the HACMP nodes, not from the Virtual I/O Server.

HACMP and virtual Ethernet

Be aware of the following considerations when implementing HACMP and virtual Ethernet:

- IP Address Takeover (IPAT) by way of aliasing must be used. IPAT by way of Replacement and MAC Address Takeover are not supported.
- Avoid using the HACMP PCI Hot Plug facility in a Virtual I/O Server environment. PCI Hot Plug operations are available through the Virtual I/O Server. When an HACMP node is using virtual I/O, the HACMP PCI Hot Plug facility is not meaningful because the I/O adapters are virtual rather than physical.
- All virtual Ethernet interfaces defined to HACMP should be treated as single-adapter networks. In particular, you must use the **ping_client_list** attribute to monitor and detect failure of the network interfaces.
- If the Virtual I/O Server has multiple physical interfaces on the same network, or if there are two or more HACMP nodes using the Virtual I/O Server in the same frame, HACMP is not informed of, and does not react to, single physical interface failures. This does not limit the availability of the entire cluster because the Virtual I/O Server routes traffic around the failure. For more information, see [Link aggregation devices](#).
- If the Virtual I/O Server has only a single physical interface on a network, failure of that physical interface is detected by HACMP. However, that failure isolates the node from the network.

Parent topic: [Client logical partitions](#)

Link aggregation devices

A link aggregation device is a group of network adapters that are grouped to provide redundancy.

A link aggregation device is a network port-aggregation technology that allows several Ethernet adapters to be aggregated, which enables them to act as a single Ethernet device. For example, `ent0` and `ent1` can be aggregated to `ent3`. The system considers these aggregated adapters as one adapter, and all adapters in the link aggregation device are given the same hardware address, so they are treated by remote systems as if they were one adapter.

In this configuration, if an adapter fails, the packets are automatically sent on the next available adapter without disruption to existing user connections. The adapter is automatically returned to service on the link aggregation device when the adapter recovers.

Parent topic: [Client logical partitions](#)

Shared Ethernet Adapter failover

Shared Ethernet Adapter failover provides redundancy by configuring a backup Shared Ethernet Adapter on a different Virtual I/O Server partition that can be used if the primary Shared Ethernet Adapter fails. The network connectivity in the client logical partitions continues without disruption.

A Shared Ethernet Adapter is comprised of a physical adapter (or several physical adapters grouped under a link aggregation device) and one or more virtual Ethernet adapters. It can provide layer 2 connectivity to multiple client logical partitions through the virtual Ethernet adapters.

The Shared Ethernet Adapter failover configuration uses the priority value given to the virtual Ethernet adapters during their creation to determine which Shared Ethernet Adapter will serve as the primary and which will serve as the backup. The Shared Ethernet Adapter that has the virtual Ethernet configured with the numerically lower priority value will be used preferentially as the primary adapter. For the purpose of communicating between themselves to determine when a failover should take place, Shared Ethernet Adapters in failover mode use a VLAN dedicated for such traffic, called the *control channel*. For this reason, a virtual Ethernet (created with a PVID that is unique on the system) must be specified as the control channel virtual Ethernet when each Shared Ethernet Adapter is created in failover mode. Using the control channel, the backup Shared Ethernet Adapter is notified when the primary adapter fails, and network traffic from the client logical partitions is sent over the backup adapter. If and when the primary Shared Ethernet Adapter recovers from its failure, it again begins actively bridging all network traffic.

A Shared Ethernet Adapter in failover mode might optionally have more than one trunk virtual Ethernet. In this case, all the virtual Ethernet adapters in a Shared Ethernet Adapter must have the same priority value. Also, the virtual Ethernet adapter used specifically for the control channel does not need to have the trunk adapter setting enabled. The virtual Ethernet adapters used for the control channel on each Shared Ethernet Adapter in failover mode must have an identical PVID value, and that PVID value must be unique in the system, so that no other virtual Ethernet adapters on the same system are using that PVID.

For a sample shared Ethernet adapter failover configuration, see [Scenario: Configuring shared Ethernet adapter failover](#).

Parent topic: [Client logical partitions](#)

Virtual I/O Server partition

Redundancy options for the Virtual I/O Server include multi-pathing, Redundant Array of Independent Disks (RAID) configurations, and link aggregation.

For information about link aggregation, see [Link aggregation devices](#).

- **Multipathing**
Multipathing for the physical storage within the Virtual I/O Server provides failover physical path redundancy and load-balancing. The multipathing solutions available in the Virtual I/O Server include MPIO as well as solutions provided by the storage vendors.
- **RAID**
Redundant Array of Independent Disks (RAID) solutions provide for device level redundancy within the Virtual I/O Server. Some RAID options, such as LVM mirroring and striping, are provided by the Virtual I/O Server software, while other RAID options are made available by the physical storage subsystem. See the VIOS datasheet for supported hardware RAID solutions.

Parent topic: [Redundancy considerations](#)

Multipathing

Multipathing for the physical storage within the Virtual I/O Server provides failover physical path redundancy and load-balancing. The multipathing solutions available in the Virtual I/O Server include MPIO as well as solutions provided by the storage vendors.

Parent topic: [Virtual I/O Server partition](#)

RAID

Redundant Array of Independent Disks (RAID) solutions provide for device level redundancy within the Virtual I/O Server. Some RAID options, such as LVM mirroring and striping, are provided by the Virtual I/O Server software, while other RAID options are made available by the physical storage subsystem. See the VIOS datasheet for supported hardware RAID solutions.

Parent topic: [Virtual I/O Server partition](#)

Security considerations

Review the security considerations for Virtual SCSI, virtual Ethernet, and Shared Ethernet adapter and the additional security options available.

A system security design plan for Virtual I/O Server involves considering the standard security features described in the following sections. Then determine if additional security controls are needed. If so, consider using the VIOS security features described in [Securing the Virtual I/O Server](#).

Architectural enhancements made in the POWER5 server system design allow cross-partition device sharing and communication. Functions such as dynamic LPAR, shared processors, virtual networking, virtual storage, and workload management all require facilities to ensure that system-security requirements are met. Cross-partition and virtualization features are designed to not introduce any security exposure beyond what is implied by the function. For example, a virtual LAN connection would have the same security considerations as a physical network connection. Carefully consider how to utilize cross-partition virtualization features in high-security environments. Any visibility between partitions must be consciously enabled through administrative system-configuration choices.

Using Virtual SCSI on the Virtual I/O Server enables the Virtual I/O Server to provide storage to client partitions. However, instead of SCSI or fiber cable, the connection for this functionality is done by the firmware. The Virtual SCSI device drivers of the Virtual I/O Server and the firmware ensure that only the system administrator of the Virtual I/O Server has control over which partitions can access data on Virtual I/O Server storage devices. For example, a client partition that has access to a logical volume `lv001` exported by the Virtual I/O Server partition cannot access `lv002`, even if it is in the same volume group.

Similar to Virtual SCSI, the firmware also provides the connection between partitions when using virtual Ethernet. The firmware provides the Ethernet switch functionality. The connection to the external network is provided by the Shared Ethernet adapter function on the Virtual I/O Server. This part of the Virtual I/O Server acts as a layer-2 bridge to the physical adapters. A VLAN ID tag is inserted into every Ethernet frame. The Ethernet switch restricts the frames to the ports that are authorized to receive frames with that VLAN ID. Every port on an Ethernet switch can be configured to be a member of several VLANs. Only the network adapters, both virtual and physical, that are connected to a port (virtual or physical) that belongs to the same VLAN can receive the frames. The implementation of this VLAN standard ensures that the partitions cannot access restricted data.

Parent topic: [Planning for the Virtual I/O Server](#)

Installing the Virtual I/O Server

Find instructions for installing the Virtual I/O Server.

The installation of the Virtual I/O Server varies, based on your environment. Installing the Virtual I/O Server where an HMC is present requires different steps than installing the Virtual I/O Server where no HMC is present. Follow these procedures based on whether or not an HMC is present.

- **[Installing the Virtual I/O Server in an HMC environment](#)**
Installing in an HMC environment involves entering the activation code, installing the Virtual I/O Server, configuring shared Ethernet adapters, and configuring virtual storage.
- **[Installing the Virtual I/O Server in a non-HMC environment](#)**
To install the Virtual I/O Server on a system that is not attached to an HMC, follow the instructions given in this topic.
- **[Connecting to the Virtual I/O Server using OpenSSH](#)**
This topic describes how to set up remote connections to the Virtual I/O Server using secure connections.

Parent topic: [Using the Virtual I/O Server](#)

Installing the Virtual I/O Server in an HMC environment

Installing in an HMC environment involves entering the activation code, installing the Virtual I/O Server, configuring shared Ethernet adapters, and configuring virtual storage.

- **Entering the activation code for Virtualization Engine technologies**
Use these instructions to enter the activation code.
- **Creating the Virtual I/O Server logical partition and partition profile**
Use these instructions to create the logical partition and partition profile.
- **Installing the Virtual I/O Server**
Installing the Virtual I/O Server includes activating the Virtualization Engine technologies, creating the partition and the partition profile, and installing the Virtual I/O Server.
- **Viewing and accepting the Virtual I/O Server license**
You must accept the license before using the Virtual I/O Server. This topic gives instructions on how to view and accept the license.

Parent topic: [Installing the Virtual I/O Server](#)

Entering the activation code for Virtualization Engine technologies

Use these instructions to enter the activation code.

If your system did not come with the Advanced POWER Virtualization feature enabled, you must use the Hardware Management Console (HMC) to enter the activation code that you received when you ordered the feature. This activation code also enables Micro-Partitioning on the system.

Note: In an Integrated Virtualization Manager environment, use the Advanced System Management Interface (ASMI) to enter your activation code. See [Enabling the Virtual I/O Server in ASMI](#).

To enter your activation code, follow these steps:

1. From the HMC, select the managed system.
2. Select Manage On Demand Activations.
3. Select Virtualization Engine Technologies.
4. Select Enter Activation Code. Type your activation code.

Parent topic: [Installing the Virtual I/O Server in an HMC environment](#)

Creating the Virtual I/O Server logical partition and partition profile

Use these instructions to create the logical partition and partition profile.

To create a partition profile, you must be a super administrator or an operator. For more information about the role of a super administrator and operator, see [Tasks and roles](#).

The Virtual I/O Server requires a minimum of 16 GB of disk space and 512 MB of memory.

To create a logical partition and a partition profile on your server using the HMC, follow these steps:

1. In the Navigation Area, open Server and Partition.
2. Select Server Management.
3. In the contents area, open the server on which you want to create the partition profile.
4. Right-click Partitions and select Create > Logical Partition.
5. Enter a name for the Virtual I/O Server partition.
6. Select the Virtual I/O Server as the Partition Environment.
7. Based on your environment, decide whether the Virtual I/O Server will be part of a workload management group.
8. Enter a profile name for the Virtual I/O Server partition.
9. Make sure that the Use all the resources in the system check box is cleared (not checked).
10. Select the appropriate amount of memory that you want to assign to the Virtual I/O Server partition. The required minimum is 512 MB.
11. Based on your environment, decide if you want to use shared or dedicated processors by making the appropriate selection.
12. Select the physical I/O resources that you want in the Virtual I/O Server partition.
13. Based on your environment, decide if the Virtual I/O Server will use I/O pools by making the appropriate selection.
14. In the Virtual I/O Adapters window, select Yes that you want to specify virtual adapters.
15. In the Create Virtual I/O Adapters window, create the appropriate adapters for your environment.
16. Based on your environment, decide if you want to specify a power-controlling partition for the Virtual I/O Server partition.
17. Decide if you want connection monitoring by making the appropriate selection.
18. If you want the Virtual I/O Server to start when the managed system starts, select the Automatically start with managed system option.
19. Select the boot mode for the Virtual I/O Server partition. In most cases, the Normal Boot Mode is the appropriate selection.
20. Verify your selections in the Profile Summary window and click Finish.

After creating your logical partition and partition profile, you must install the Virtual I/O Server. For installation procedures, see [Installing the Virtual I/O Server](#).

Parent topic: [Installing the Virtual I/O Server in an HMC environment](#)

Installing the Virtual I/O Server

Installing the Virtual I/O Server includes activating the Virtualization Engine technologies, creating the partition and the partition profile, and installing the Virtual I/O Server.

You can install the Virtual I/O Server either from the HMC or from a CD drive that is attached to the Virtual I/O Server partition.

- **[Installing the Virtual I/O Server from the HMC](#)**
Find instructions for installing the Virtual I/O Server from the HMC by using the **installios** command.
- **[Installing the Virtual I/O Server from CD or DVD](#)**
Find instructions for installing the Virtual I/O Server from a CD or DVD device that is attached to the Virtual I/O Server partition.

Parent topic: [Installing the Virtual I/O Server in an HMC environment](#)

Installing the Virtual I/O Server from the HMC

Find instructions for installing the Virtual I/O Server from the HMC by using the **installios** command.

In the procedure, you will install the Virtual I/O Server from the HMC by using the **installios** command.

Prerequisites

This procedure assumes that there is an HMC attached to the managed system, that the Virtual I/O Server logical partition and partition profile have been created, and that the partition has at least one Ethernet adapter and a 16 GB disk assigned to it. If you need to create the logical partition and partition profile, see [Creating the Virtual I/O Server logical partition and partition profile](#).

By default, the **installios** command installs the Virtual I/O Server through the public network interface. To install the Virtual I/O Server through a private network interface, the `INSTALLIOS_PRIVATE_IF` environment variable must be used. An example is given in the instructions that follow.

To complete this procedure, you need to have `hmcsuperadmin` authority.

Before beginning this procedure, have the following information ready:

- Static IP address for the Virtual I/O Server
- Subnet mask for the Virtual I/O Server
- Default gateway for the Virtual I/O Server

To install the Virtual I/O Server, follow these steps:

1. Insert the Virtual I/O Server CD or DVD into the HMC.
2. If you are installing the Virtual I/O Server through the public network interface, continue to step 3. If you are installing the Virtual I/O Server through a private network interface, type the following from the HMC command line:

```
export INSTALLIOS_PRIVATE_IF=interface
```

where *interface* is the network interface through which the installation should take place.

3. From the HMC command line, type:

```
installios
```

4. Follow the installation instructions according to the system prompts.
5. Check for updates to the Virtual I/O Server.

The Virtual I/O Server logical partition is now ready to be configured and managed. The following tasks can now be performed:

- After the installation is complete, the only active user ID is the prime administrator (`padmin`). Create user IDs using the **mkuser** command. For information about user types, see [User types](#).
- Configure the TCP/IP connection for the Virtual I/O Server. To configure the network settings, use the **mktcpip** command .

Important: This task must be completed before you can perform any dynamic LPAR operations.

- Configure Virtual SCSI and shared Ethernet resources. For instructions, see [Managing the Virtual I/O Server](#)

Parent topic: [Installing the Virtual I/O Server](#)

Installing the Virtual I/O Server from CD or DVD

Find instructions for installing the Virtual I/O Server from a CD or DVD device that is attached to the Virtual I/O Server partition.

In this procedure, you install the Virtual I/O Server using the logical partition's optical device.

Prerequisites

This procedure assumes that there is an HMC attached to the system. Before you begin this procedure, you should have already used the HMC to create a Virtual I/O Server logical partition and partition profile. If you have not yet created the Virtual I/O Server logical partition, see [Creating the Virtual I/O Server logical partition and partition profile](#). In addition, a CD/DVD optical device must be assigned to the Virtual I/O Server logical partition.

To install the Virtual I/O Server from CD or DVD, follow these steps:

1. Activate the Virtual I/O Server logical partition:
 - a. Insert the Virtual I/O Server CD or DVD into the Virtual I/O Server logical partition.
 - b. On the HMC, right-click the partition to open the menu.
 - c. Click Activate. The Activate Partition menu opens with a selection of partition profiles. Ensure the correct profile is highlighted.
 - d. Select Open a terminal window or console session to open a virtual terminal (vterm) window.
 - e. Click (Advanced...) to open the advanced options menu.
 - f. For the boot mode, select SMS.
 - g. Click OK to close the advanced options menu.
 - h. Click OK. A virtual terminal window opens for the partition.
2. Select the boot device:
 - a. Select Select Boot Options and press Enter.
 - b. Select Select Install/Boot Device and press Enter.
 - c. Select Select 1st Boot Device and press Enter.
 - d. Select CD/DVD and press Enter.
 - e. Select the media type that corresponds to the optical device and press Enter.
 - f. Select the device number that corresponds to the optical device and press Enter.
 - g. Set the boot sequence to configure the first boot device. The optical device is now the first device in the Current Boot Sequence list.
 - h. Exit the SMS menu by pressing the `x` key, and confirm that you want to exit SMS.
3. Install the Virtual I/O Server:
 - a. Select the desired console and press Enter.
 - b. Select a language for the BOS menus and press Enter.
 - c. Select Start Install Now with Default Settings and press Enter.
 - d. Select Continue with Install. The system will reboot after the installation is complete.
4. Accept the license agreement by following the steps in [Viewing and accepting the Virtual I/O Server license](#).
5. Check for updates to the Virtual I/O Server.

The Virtual I/O Server logical partition is now ready to be configured and managed. The following tasks can now be performed:

- After the installation is complete, the only active user ID is the prime administrator (padmin). Create user IDs using the `mkuser` command. For information about user types, see [User types](#).
- Configure the TCP/IP connection for the Virtual I/O Server. To configure the network settings, use the `mktcpip` command .

Note: This task must be completed before you can perform any dynamic LPAR operations.

- Configure Virtual SCSI and shared Ethernet resources. For instructions, see [Managing the Virtual I/O Server](#)

Parent topic: [Installing the Virtual I/O Server](#)

Viewing and accepting the Virtual I/O Server license

You must accept the license before using the Virtual I/O Server. This topic gives instructions on how to view and accept the license.

This task assumes that Virtual I/O Server partition profile has been created and that the Virtual I/O Server is installed in that partition.

1. Log in to the Virtual I/O Server using `padmin` as the user ID.
2. Choose a new password.
3. In the installation program, English is the default language. If you need to change the language setting for the system, follow these steps. Otherwise, proceed to step 4.
 - a. View the available languages by typing the following command:

```
chlang -ls
```

- b. Change the language by typing the following command, replacing *name* with the name of the language you are switching to:

```
chlang -lang Name
```

Note: If the language fileset is not installed, use the **-dev Media** flag to install it.

For example, to install and change the language to Japanese, type the following command:

```
chlang -lang ja_JP -dev /dev/cd0
```

4. View the license by typing `license -ls` on the command line. By default, the license is displayed in English. To change the language in which the license is displayed, follow these steps:
 - a. View the list of available locales to display the license by typing the following:

```
license -ls
```

- b. View the license in another language by typing the following command:

```
license -view -lang Name
```

For example, to view the license in Japanese, type the following:

```
license -view -lang ja_JP
```

5. Accept the license agreement for the Virtual I/O Server by typing `license -accept -lang Name`.
6. Check for updates to the Virtual I/O Server.

The Virtual I/O Server logical partition is now ready to be configured and managed. The following tasks can now be performed:

- After the installation is complete, the only active user ID is the prime administrator (`padmin`). You can create user IDs by using the `mkuser` command. For information about user types, see [User types](#).
- Configure the TCP/IP connection for the Virtual I/O Server. To configure the network settings, use the `mktcpip` command .

Important: This task must be completed before you can perform any dynamic LPAR operations.

- Configure Virtual SCSI and shared Ethernet resources. For instructions, see [Managing the Virtual I/O Server](#)

Parent topic: [Installing the Virtual I/O Server in an HMC environment](#)

Installing the Virtual I/O Server in a non-HMC environment

To install the Virtual I/O Server on a system that is not attached to an HMC, follow the instructions given in this topic.

When the Virtual I/O Server is installed in a non-HMC environment, the Integrated Virtualization Manager is installed. In addition to the command-line interface, the Integrated Virtualization Manager provides a graphical user interface that can be used to create partitions and manage resources on the system.

Prerequisites

This installation procedure assumes that the system has been set up with a physical serial console and that the Advanced POWER Virtualization activation code has been entered using ASMI. If any of these steps have not been performed, see [Initial server setup](#) for additional instructions and checklists.

1. Power on the system, and enter the SMS menu by pressing and holding the F1 key.
2. Insert the Virtual I/O Server CD or DVD into the optical drive.
3. Select the CD or DVD as the boot device:
 - a. Select Select Boot Options and press Enter.
 - b. Select Select Install/Boot Device and press Enter.
 - c. Select Select 1st Boot Device and press Enter.
 - d. Select CD/DVD and press Enter.
 - e. Select the media device that corresponds to the optical device and press Enter.
 - f. Select the device number that corresponds to the optical device and press Enter.
 - g. Set the boot sequence to configure the first boot device. The optical device is now the first device in the Current Boot Sequence list.
 - h. Exit the SMS menu by pressing the \times key, and confirm that you want to exit SMS.
4. Install the Virtual I/O Server:
 - a. Select the desired console and press Enter.
 - b. Select a language for the BOS menus and press Enter.
 - c. Select Start Install Now with Default Settings.
 - d. Select Continue with Install. The system restarts after the installation is complete.
5. Accept the license agreement by following the steps in [Viewing and accepting the Virtual I/O Server license](#).
6. Check for updates to the Virtual I/O Server.

The Virtual I/O Server is now installed on your system. You can now manage the system using either the Virtual I/O Server command-line interface or using the Integrated Virtualization Manager graphical user interface. For more information, see [Virtual I/O Server command-line interface](#), [Partitioning with Integrated Virtualization Manager](#), and [Managing Integrated Virtualization Manager](#).

Parent topic: [Installing the Virtual I/O Server](#)

Connecting to the Virtual I/O Server using OpenSSH

This topic describes how to set up remote connections to the Virtual I/O Server using secure connections.

You can use the Open Source Secure Sockets Layer (OpenSSL) and Portable Secure Shell (OpenSSH) software to connect to the Virtual I/O Server (VIOS) using secure connections. However, to connect to the VIOS using OpenSSL and Portable OpenSSH, you must download and install these tools in the Virtual I/O Server management partition. For more information about OpenSSL and Portable OpenSSH, see <http://www.openssl.org/> and <http://openssh.org/portable.html>.

Downloading the Open Source software

The OpenSSL software contains the encrypted library that is required to use the OpenSSH software. This software is available in RPM packages from the [AIX Toolbox for Linux Applications](#). The OpenSSH software is available from the [SourceFORGE.net](#) Web site, and is also available in the AIX Expansion Pack. Use the following tasks to download the software:

1. Download the OpenSSL RPM package to your workstation or download host computer.
 - a. To get the RPM package, go to the following Web site: <http://www.ibm.com/servers/aix/products/aixos/linux/download.html> and click the **AIX Toolbox Cryptographic Content** link on the right side of the Web page.
 - b. If you are not registered to download the RPM packages, complete the registration process and accept the license agreement. After registering (or signing in), you are automatically redirected to the download page.
 - c. Select the following package for download: openssl - Secure Sockets Layer and cryptography libraries and tools and click the Download Now button to start the download.
2. Download the OpenSSH software or install the software from the AIX Expansion Pack. To download the software, complete the following steps:
 - a. From your workstation (or download host computer), go to the following Web site: <https://sourceforge.net/projects/openssh-aix>.
 - b. Click Download OpenSSH on AIX to view the latest file releases.
 - c. Select the appropriate download package and click Download.
 - d. Click the openssh package (tar.Z file) to continue with the download.
3. Create a directory on the Virtual I/O Server for the Open Source software files and transfer the software packages. For example, to create an installation directory named `install_ssh`, issue the following command: `mkdir install_ssh`. You can use ftp to transfer the software packages to the VIOS. Make sure the ftp server is started on the VIOS by issuing the following command: `startnetsvc ftp`. To transfer the files, issue the following ftp commands from the computer on which you downloaded the software packages:
 - a. Open up an ftp session to the VIOS on your local host: `ftp vios_server_hostname`
 - b. At the ftp prompt, change to the installation directory you created for the Open Source files: `cd install_ssh`
 - c. Set the transfer mode to binary: `binary` (or `bin`)
 - d. Turn off interactive prompting (if set on): `prompt`
 - e. Transfer the downloaded software to the VIOS: `mput ssl_software_pkg`.
 - f. Close the ftp session (after transferring both software packages): `quit`

Install the Open Source software on the Virtual I/O Server

To install the Open Source software on the VIOS, issue the following command from the VIOS command line:

```
updateios -dev install_ssh -accept -install
```

When the installation is complete, the installation program automatically starts the Secure Shell daemon (sshd) on the server. You can begin using the **ssh** and **scp** commands; no further configuration is required.

Note: The **sftp** command is not supported by the Virtual I/O Server. Non-interactive shells are not currently supported using OpenSSH with the Virtual I/O Server.

Parent topic: [Installing the Virtual I/O Server](#)

Configuration scenarios for the Virtual I/O Server

The following scenarios show examples of networking configurations for the Virtual I/O Server logical partition and the client logical partitions. Use the following scenarios and configuration examples to understand more about the Virtual I/O Server and its components.

- **Scenario: Configuring a Virtual I/O Server without VLAN tagging**
Use this scenario to help you become familiar with creating a network without VLAN tagging.
- **Scenario: Configuring a Virtual I/O Server using VLAN tagging**
Use this scenario to help you become familiar with creating a network using VLAN tagging.
- **Scenario: Configuring shared Ethernet adapter failover**
Use this article to help you become familiar with typical shared Ethernet adapter failover scenario.
- **Scenario: Configuring Network Interface Backup in Virtual I/O clients without VLAN tagging**
Use this scenario to become familiar with using a Network Interface Backup configuration in Virtual I/O clients that are running AIX partitions and are not configured for VLAN tagging.
- **Scenario: Configuring Multi-Path I/O for AIX client logical partitions**
Multi-Path I/O (MPIO) helps provide increased availability of virtual SCSI resources by providing redundant paths to the resource. This topic describes how to set up Multi-Path I/O for AIX client logical partitions.

Parent topic: [Using the Virtual I/O Server](#)

Scenario: Configuring a Virtual I/O Server without VLAN tagging

Use this scenario to help you become familiar with creating a network without VLAN tagging.

Situation

You are the system administrator responsible for planning and configuring the network in an environment with the Virtual I/O Server running. You want to configure a single logical subnet on the system that communicates with the switch.

Objective

The objective of this scenario is to configure the network where only Port Virtual LAN ID (PVID) is used, the packets are not tagged, and a single internal network is connected to a switch. There are no virtual local area networks (VLAN) tagged ports set up on the Ethernet switch, and all virtual Ethernet adapters are defined using a single default PVID and no additional VLAN IDs (VIDs).

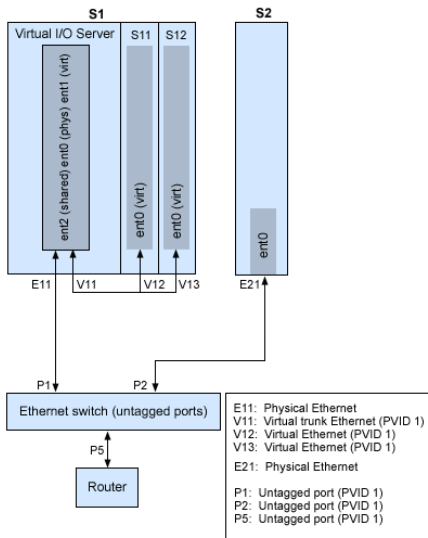
Prerequisites and assumptions

- The [Hardware Management Console \(HMC\)](#) was set up using the following tasks:
 - ◆ The HMC was cabled, as described in [Cabling the HMC](#).
 - ◆ You completed the planning process for [Gathering required configuration settings](#), and you understand how you want to configure your HMC.
 - ◆ You used the [Guided Setup wizard](#) or the [HMC configuration checklist](#) to set up the HMC.
- You understand the [concepts](#) for partitioning the server.
- The Virtual I/O Server partition has been created and the Virtual I/O Server has been installed. See the instructions in [Installing the Virtual I/O Server in an HMC environment](#).
- You have created the remaining logical partitions that you want added to the network configuration.
- You have an Ethernet switch and a router ready to add to the configuration.
- You have IP addresses for all logical partitions and systems that will be added to the configuration.

While this procedure describes configuration in an HMC environment, this configuration is also possible in an Integrated Virtualization Manager environment.

Configuration steps

The following figure shows the configuration that will be completed during this scenario.



Using the preceding figure as a guide, follow these steps:

1. Set up an Ethernet switch with untagged ports. Alternatively, you can use an Ethernet switch that does not use VLAN.
2. For system S1, use the HMC to create a virtual Ethernet adapter (V11) for the Virtual I/O Server with the trunk setting, PVID set to 1, and no additional VIDs.
3. For system S1, use the HMC to create virtual Ethernet adapters V12 and V13 for partitions S11 and S12, respectively, with PVID set to 1 and no additional VIDs.
4. For system S1, use the HMC to assign physical Ethernet adapter E11 to the Virtual I/O Server and connect the adapter to the Ethernet switch port P1.
5. On the Virtual I/O Server, set up shared Ethernet adapter ent2 with the physical adapter ent0 and virtual adapter ent1.
6. Start the logical partitions. The process recognizes the virtual devices that were created in Step 1.
7. Configure IP addresses for S11 (en0), S12 (en0), and S2 (en0), so that they all belong to the same subnet with the router connected to Ethernet switch port Power5.

The shared Ethernet adapter on the Virtual I/O Server partition can also be configured with IP addresses on the same subnet. This is required only for network connectivity to the Virtual I/O Server logical partition.

Parent topic: [Configuration scenarios for the Virtual I/O Server](#)

Scenario: Configuring a Virtual I/O Server using VLAN tagging

Use this scenario to help you become familiar with creating a network using VLAN tagging.

Situation

You are the system administrator responsible for planning and configuring the network in an environment with the Virtual I/O Server running. You would like to configure the network so that two logical subnets exist, with some partitions on each subnet.

Objective

The objective of this scenario is to configure multiple networks to share a single physical Ethernet adapter. Systems on the same subnet are required to be on the same VLAN and therefore have the same VLAN ID, which allows communication without having to go through the router. The separation in the subnets is achieved by ensuring that the systems on the two subnets have different VLAN IDs.

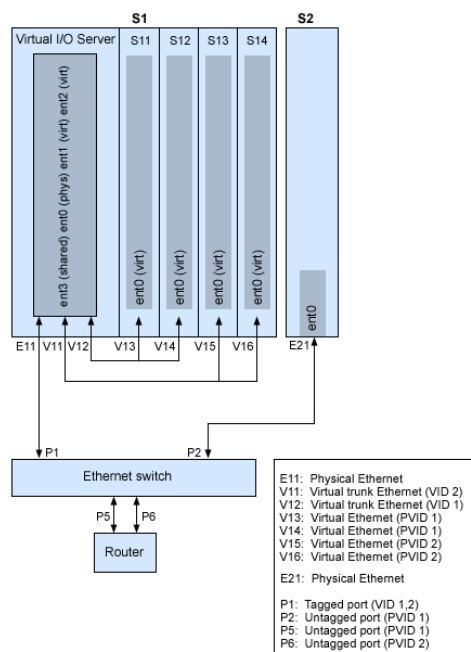
Prerequisites and assumptions

- The [Hardware Management Console \(HMC\)](#) was set up.
 - ◆ The HMC was [cabled](#).
 - ◆ You completed the planning process and you understand how you want to [configure](#) your HMC.
 - ◆ You used the [Guided Setup wizard](#) or the [HMC configuration checklist](#) to set up the HMC.
- You understand the [concepts](#) for partitioning the server.
- The Virtual I/O Server partition has been created and the Virtual I/O Server has been installed. See the instructions in [Installing the Virtual I/O Server in an HMC environment](#).
- You have created the remaining logical partitions that you want added to the network configuration.
- You have an Ethernet switch and a router ready to add to the configuration.
- You have IP addresses for all partitions and systems that will be added to the configuration.

You cannot use VLAN in an Integrated Virtualization Manager environment.

Configuration steps

The following figure shows the configuration that will be completed during this scenario.



Using the preceding figure as a guide, follow these steps.

1. Set up the Ethernet switch ports as follows:
 - ◆ P1: Tagged port (VID 1, 2)
 - ◆ P2: Untagged port (PVID 1)
 - ◆ P5: Untagged port (PVID 1)
 - ◆ P6: Untagged port (PVID 2)

For instructions on configuring the ports, see the documentation for your switch.
2. For system S1, use the HMC to create virtual Ethernet adapters for the Virtual I/O Server:
 - ◆ Create virtual Ethernet adapter V11 for the Virtual I/O Server with the trunk setting selected and VID set to 2. Specify an unused PVID value. This value is required, even though it will not be used.
 - ◆ Create virtual Ethernet adapter V12 for the Virtual I/O Server with the trunk setting selected and VID set to 1. Specify an unused PVID value. This value is required, even though it will not be used.

3. For system S1, use the HMC to create virtual Ethernet adapters for other partitions:
 - ◆ Create virtual adapters V13 and V14 for partitions S11 and S12, respectively, with PVID set to 2 and no additional VIDs.
 - ◆ Create virtual adapters V15 and V16 for partitions S13 and S14, respectively, with PVID set to 1 and no additional VIDs.
4. For system S1, use the HMC to assign the physical Ethernet adapter (E11) to the Virtual I/O Server and connect the adapter to the Ethernet switch port P1.
5. Using the Virtual I/O Server command-line interface, set up a shared Ethernet adapter ent3 with the physical adapter ent0 and virtual adapters ent1 and ent2.
6. Configure IP addresses for the following:
 - ◆ S13 (en0), S14 (en0), and S2 (en0) belong to VLAN 1 and are on the same subnet. The router is connected to Ethernet switch port Power5.
 - ◆ S11 (en0) and S12 (en0) belong to VLAN 2 and are on the same subnet. The router is connected to Ethernet switch port P6.

You can configure the shared Ethernet adapter on the Virtual I/O Server partition with an IP address. This is required only for network connectivity to the Virtual I/O Server.

As the tagged VLAN network is being used, you must define additional VLAN devices over the shared Ethernet adapters before configuring IP addresses.

Parent topic: [Configuration scenarios for the Virtual I/O Server](#)

Scenario: Configuring shared Ethernet adapter failover

Use this article to help you become familiar with typical shared Ethernet adapter failover scenario.

Situation

You are the system administrator responsible for planning and configuring the network in an environment with the Virtual I/O Server running. You want to provide higher network availability to the client logical partition on the system. This can be accomplished by configuring a backup shared Ethernet adapter in a different Virtual I/O Server partition.

Objective

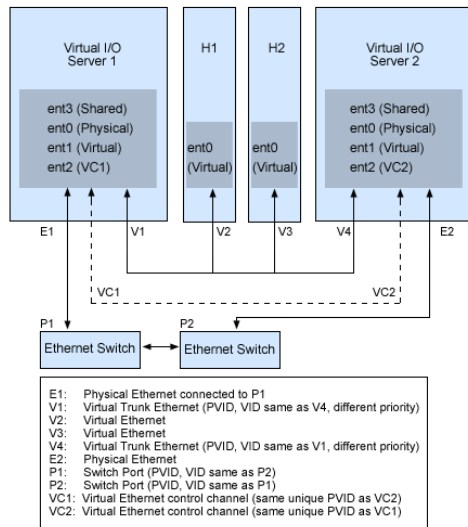
The objective of this scenario is to configure primary and backup shared Ethernet adapters in the Virtual I/O Server logical partitions so that network connectivity in the client partitions will not be lost in the case of adapter failure.

Prerequisites and assumptions

- The [Hardware Management Console](#) (HMC) was set up.
 - ◆ The HMC was [cabled](#).
 - ◆ You completed the planning process and you understand how you want to [configure](#) your HMC.
 - ◆ You used the [Guided Setup wizard](#) or the [HMC configuration checklist](#) to set up the HMC.
- You understand the [concepts](#) for partitioning the server.
- Two separate Virtual I/O Server partitions have been created and the Virtual I/O Server has been installed in each. See the instructions in [Installing the Virtual I/O Server in an HMC environment](#).
- You understand what shared Ethernet adapter failover and how it works. See [Shared Ethernet Adapter failover](#).
- You have created the remaining logical partitions that you want added to the network configuration.
- Each Virtual I/O Server partition has an available physical Ethernet adapter assigned to it.
- You have IP addresses for all partitions and systems that will be added to the configuration.

You cannot use the Integrated Virtualization Manager with multiple Virtual I/O Server partitions on the same server.

The following image depicts a configuration where the shared Ethernet adapter (SEA) failover feature is set up. The client partitions H1 and H2 are accessing the physical network using the shared Ethernet adapters, which are the primary adapters. The virtual Ethernet adapters used in the shared Ethernet setup are configured with the same VLAN membership information (PVID, VID), but have different priorities. A dedicated virtual network forms the control channel and is required to facilitate communication between the primary and backup shared Ethernet device.



Using the preceding figure as a guide, follow these steps:

- On the HMC, create the virtual Ethernet adapters following these guidelines:
 - Configure the virtual adapters to be used for data as trunk adapters by selecting the trunk setting.
 - Assign different prioritization values (valid values are 1-15) to each virtual adapter.
 - Configure another virtual Ethernet to be used for the control channel by giving it a unique PVID value. Make sure you use the same PVID when creating this virtual Ethernet for both Virtual I/O Server partitions.
- Using the Virtual I/O Server command line, run the following command to configure the shared Ethernet adapter. Run this command on both Virtual I/O Server partitions involved in the configuration:

```
mkvdev -sea physical_adapter -vadapter virtual_adapter -default virtual_adapter\  
-defaultid PVID_of_virtual_adapter -attr ha_mode=auto ctl_chan=control_channel_adapter
```

For example, in this scenario, we ran the following command on both Virtual I/O Server partitions:

```
mkvdev -sea ent0 -vadapter ent1 -default ent1 -defaultid 60 -attr ha_mode=auto ctl_chan=ent2
```

Parent topic: [Configuration scenarios for the Virtual I/O Server](#)

Scenario: Configuring Network Interface Backup in Virtual I/O clients without VLAN tagging

Use this scenario to become familiar with using a Network Interface Backup configuration in Virtual I/O clients that are running AIX partitions and are not configured for VLAN tagging.

Situation

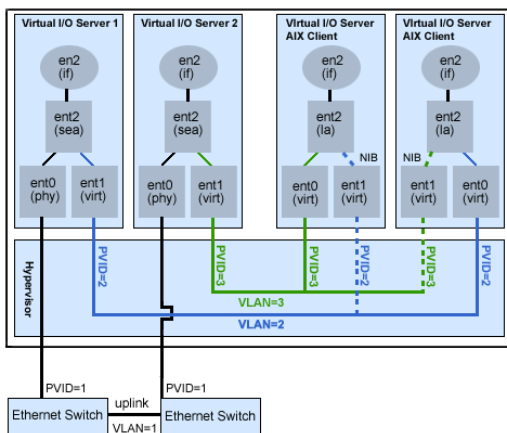
In this scenario, you want to configure a highly available virtual environment for your bridged network using the Network Interface Backup (NIB) approach to access external networks from your Virtual I/O clients. You do not plan to use VLAN tagging in your network setup. This approach requires you to configure a second Ethernet adapter on a different VLAN for each client and requires a Link Aggregation adapter with NIB

features. This configuration is available for AIX partitions.

Typically, an SEA failover configuration is the recommended configuration for most environments because it supports environments with or without VLAN tagging. Also, the NIB configuration is more complex than an SEA failover configuration because it must be implemented on each of the Virtual I/O clients. However, SEA failover was not available prior to version 1.2 of Virtual I/O Server, and NIB was the only approach to a highly available virtual environment. Also, you might consider that in an NIB configuration you can distribute clients over both Shared Ethernet adapters (SEAs) in such a way that half of them will use the first SEA and the other half will use the second SEA as primary adapter.

Objective

Create a virtual Ethernet environment using a Network Interface Backup configuration as depicted in the following figure:



Prerequisites and assumptions

Before completing the configuration tasks, review the following prerequisites and assumptions.

- The [Hardware Management Console \(HMC\)](#) is already set up. For setup instructions, see [Setting up the HMC](#).
- Two separate Virtual I/O Server partitions have been created and the Virtual I/O Server has been installed in each. See the instructions in [Installing the Virtual I/O Server in an HMC environment](#).
- You have created the remaining logical partitions that you want added to the network configuration.
- Each Virtual I/O Server partition has an available physical Ethernet adapter assigned to it.
- You have IP addresses for all partitions and systems that will be added to the configuration.

Configuration tasks

Using the figure as a guide, complete the following tasks to configure the NIB virtual environment.

1. Create a LAN connection between the Virtual I/O Servers and the external network:
 - a. Configure a Shared Ethernet adapter on the primary Virtual I/O Server that bridges traffic between the virtual Ethernet and the external network. See [Configuring the shared Ethernet adapter](#).
 - b. Configure a Shared Ethernet adapter on the second Virtual I/O Server, as in step 1.
2. For each client partition, use the HMC to create a virtual Ethernet whose PVID matches the PVID of the primary Virtual I/O Server. This will be used as the primary adapter.
3. For each client partition, use the HMC to create a second virtual Ethernet whose PVID matches the PVID of the second (backup) Virtual I/O Server. This will be used as the backup adapter.
4. Create the Network Interface Backup setup using an EtherChannel/Link Aggregation configuration. To create this configuration, follow the instructions for Configuring an EtherChannel in the AIX publication, [AIX System Management Guide: Communications and Networks](#) making sure you specify the following items:
 - a. Select the primary Ethernet Adapter.
 - b. Select the Backup Adapter.

- c. Specify the Internet Address to Ping. Select the IP address or hostname of a host outside of the Virtual I/O Server system that NIB will continuously ping to detect VIOS failure.

Note: Keep in mind, when configuring NIB with two virtual Ethernet adapters, the internal networks used must stay separated in the POWER hypervisor, so you must use different PVIDs for the two adapters in the client and cannot use additional VIDs on them.

Parent topic: [Configuration scenarios for the Virtual I/O Server](#)

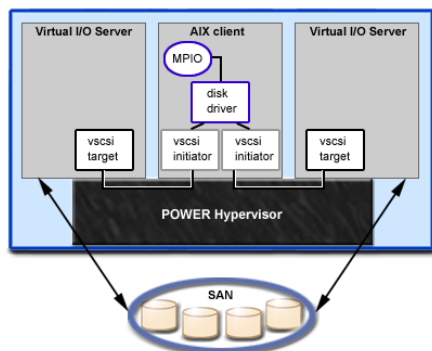
Scenario: Configuring Multi-Path I/O for AIX client logical partitions

Multi-Path I/O (MPIO) helps provide increased availability of virtual SCSI resources by providing redundant paths to the resource. This topic describes how to set up Multi-Path I/O for AIX client logical partitions.

In order to provide MPIO to AIX client logical partitions, you must have two Virtual I/O Server logical partitions configured on your system. This procedure assumes that the disks are already allocated to both the Virtual I/O Server logical partitions involved in this configuration.

To configure MPIO, follow these steps. In this scenario, hdisk5 in the first Virtual I/O Server logical partition, and hdisk7 in the second Virtual I/O Server logical partition, are used in the configuration.

The following figure shows the configuration that will be completed during this scenario.



Using the preceding figure as a guide, follow these steps:

1. Using the HMC, create SCSI server adapters on the two Virtual I/O Server logical partitions.
2. Using the HMC, create two virtual client SCSI adapters on the client logical partitions, each mapping to one of the Virtual I/O Server logical partitions.
3. On either of the Virtual I/O Server logical partitions, determine which disks are available by typing `lsdev -type disk`. Your results look similar to the following:

name	status	description
hdisk3	Available	MPIO Other FC SCSI Disk Drive
hdisk4	Available	MPIO Other FC SCSI Disk Drive
hdisk5	Available	MPIO Other FC SCSI Disk Drive

Select which disk that you want to use in the MPIO configuration. In this scenario, we selected hdisk5.

4. Determine the PVID of the disk that you have selected using the **lspv** command. In this scenario, we typed `lspv hdisk5`. Your results look similar to the following:

```
hdisk5          00c3e35ca560f919          None
```

The second value is the PVID. In this scenario, the PVID is 00c3e35ca560f919. Note this value.

5. List the attributes of the disk using the **lsdev** command. In this scenario, we typed `lsdev -dev hdisk5 -attr`. Your results look similar to the following

```
..
algorithm      fail_over          Algorithm          True
..
lun_id         0x5463000000000000 Logical Unit Number ID False
..
..
pvid          00c3e35ca560f9190000000000000000 Physical volume identifier False
..
reserve_policy single_path        Reserve Policy     True
```

Note the values for `lun_id` and `reserve_policy`. If the `reserve_policy` attribute is set anything other than `no_reserve`, then you must change it. Set the `reserve_policy` to `no_reserve` by typing `chdev -dev hdiskx -attr reserve_policy=no_reserve`.

6. On the second Virtual I/O Server logical partition, list the physical volumes by typing `lspv`. In the output, locate the disk that has the same PVID as the disk identified previously. In this scenario, the PVID for `hdisk7` matched:

```
hdisk7          00c3e35ca560f919          None
```

Tip: Although the PVID values should be identical, the disk numbers on the two Virtual I/O Server logical partitions might vary.

7. Determine if the `reserve_policy` attribute is set to `no_reserve` using the **lsdev** command. In this scenario, we typed `lsdev -dev hdisk7 -attr`. You see results similar to the following:

```
algorithm      fail_over          Algorithm          True
..
lun_id         0x5463000000000000 Logical Unit Number ID False
..
..
pvid          00c3e35ca560f9190000000000000000 Physical volume identifier False
..
reserve_policy single_path        Reserve Policy
```

If the `reserve_policy` attribute is set anything other than `no_reserve`, you must change it. Set the `reserve_policy` to `no_reserve` by typing `chdev -dev hdiskx -attr reserve_policy=no_reserve`.

8. On both Virtual I/O Server logical partitions, use the **mkvdev** to create the virtual devices. In each case, use the appropriate `hdisk` value. In this scenario, we type the following commands:
- ◆ On the first Virtual I/O Server logical partition, we typed `mkvdev -vdev hdisk5 -vadapter vhost5 -dev vhdisk5`
 - ◆ On the second Virtual I/O Server logical partition, we typed `mkvdev -vdev hdisk7 -vadapter vhost7 -dev vhdisk7`

The same LUN is now exported to the client logical partition from both Virtual I/O Server logical partitions

9. AIX can now be installed on the client logical partition. For instructions on installing AIX, see [Installing AIX in a Partitioned Environment](#).
10. After you have installed AIX on the client logical partition, check for MPIO by doing the following:
- a. `lsdev -Cc disk`

You see results similar to the following:

```
hdisk0 Available Virtual SCSI Disk Drive
```

- b. `lspv`

You see results similar to the following:

```
hdisk0          00c3e35ca560f919          rootvg          active
```

- c. `lspath`

You see results similar to the following:

```
Enabled hdisk0 vscsi0
Enabled hdisk0 vscsi1
```

If one of the Virtual I/O Server logical partitions fails, the result of the **lspath** command look similar to the following:

```
Failed hdisk0 vscsi0
Enabled hdisk0 vscsi1
```

Unless the `hcheck_mode` and `hcheck_interval` attributes are set, the state will continue to show `Failed` even after the disk has recovered. To have the state updated automatically, type `chdev -l hdiskx -a hcheck_interval=60 -P`. The client logical partition must be rebooted for this change to take effect.

Parent topic: [Configuration scenarios for the Virtual I/O Server](#)

Securing the Virtual I/O Server

Understand the concepts for securing your Virtual I/O Server environment.

The Virtual I/O Server (VIOS) provides extra security features that enable you to control access to the virtual environment and ensure the security of your system. These features are available with Virtual I/O Server version 1.3 or later. The following topics discuss the security features available and provide tips for ensuring a secure environment for your Virtual I/O Server setup.

- [Introduction to Virtual I/O Server security](#)
Become familiar with the Virtual I/O Server security features.
- [Configuring Virtual I/O Server system security hardening](#)
Set the security level to specify security hardening rules for your Virtual I/O Server (VIOS) system.
- [Configuring VIOS firewall settings](#)
Enable the Virtual I/O Server (VIOS) firewall to control IP activity.

Parent topic: [Using the Virtual I/O Server](#)

Introduction to Virtual I/O Server security

Become familiar with the Virtual I/O Server security features.

Beginning with version 1.3 of the Virtual I/O Server, you can set security options that provide tighter security controls over your Virtual I/O Server environment. These options allow you to select a level of system security hardening and specify the settings allowable within that level. The Virtual I/O Server security feature also allows you to control network traffic by enabling the Virtual I/O Server firewall. You can configure these options using the `viorecure` command.

The `viorecure` command enables you to set, change, and view current security settings. The settings are not enabled by default, you must issue the `viorecure` command to specify the options. For more information about this command, see the [viorecure command](#) in the [Virtual I/O Server Commands Reference](#)

The following sections provide an overview of these features.

VIOS system security hardening

The system security hardening feature protects all elements of a system by tightening security or implementing a higher level of security. Although hundreds of security configurations are possible with the VIOS security settings, you can easily implement security controls by specifying a high, medium, or low security level. For information about configuring security levels, see the [Configuring Virtual I/O Server system security hardening](#) topic.

The system security hardening features provided by Virtual I/O Server enable you to specify values such as the following:

- password policy settings
- `usrck`, `pwdck`, `grpck`, and `sysck` actions
- Default file creation settings
- crontab settings

Configuring a system at too high a security level might deny services that are needed. For example, `telnet` and `rlogin` are disabled for high level security because the login password is sent over the network unencrypted. If a system is configured at too low a security level, the system might be vulnerable to security threats. Since each enterprise has its own unique set of security requirements, the predefined High, Medium, and Low security configuration settings are best suited as a starting point for security configuration rather than an exact match for the security requirements of a particular enterprise. As you become more familiar with the security settings, you can make adjustments by choosing the hardening rules you want to apply. You can get information about the hardening rules by running the `man` command.

VIOS firewall

The Virtual I/O Server firewall enables you to enforce limitations on IP activity in your virtual environment. With this feature, you can specify which ports and or network services are allowed access to the Virtual I/O Server system. For example, if you need to restrict login activity from an unauthorized port, you can specify the port name or number and specify `deny` to remove it from the allow list. You can also restrict a specific IP address.

Before configuring firewall settings, you must first enable it. For information about enabling the VIOS firewall and configuring firewall settings see the topic, [Configuring VIOS firewall settings](#).

Parent topic: [Securing the Virtual I/O Server](#)

Configuring Virtual I/O Server system security hardening

Set the security level to specify security hardening rules for your Virtual I/O Server (VIOS) system.

To implement system security hardening rules, you can use the `viosecure` command to specify a security level of high, medium, or low. A default set of rules is defined for each level. You can also set a level of default, which returns the system to the system standard settings and removes any level settings that have been applied.

The low level security settings are a subset of the medium level security settings, which are a subset of the high level security settings. Therefore, the *high* level is the most restrictive and provides the greatest level of control. You can apply all of the rules for a specified level or select which rules to activate for your environment. By default, no VIOS security levels are set; you must run the `viosecure` command to enable the settings.

Use the following tasks to configure the system security settings:

Setting a security level

To set a VIOS security level of high, medium, or low, use the command `viosecure -level`, as in the following example:

```
viosecure -level low -apply
```

Changing the settings in a security level

To set a VIOS security level in which you specify which hardening rules to apply for the setting, run the `viosecure` command interactively, as in the following example:

1. At the VIOS command line, type `viosecure -level high`. All the security level options (hardening rules) at that level are displayed ten at a time (pressing Enter displays the next set in the sequence).
2. Review the options displayed and make your selection by entering the numbers, separated by a comma, that you want to apply, or type ALL to apply all the options or NONE to apply none of the options.
3. Press Enter to display the next set of options, and continue entering your selections.

Note: To exit the command without making any changes, type "q".

Viewing the current security setting

To display the current VIOS security level setting use the `viosecure` command with the `-view` flag, as in the following example:

```
viosecure -view
```

Removing security level settings

To unset any previously set system security levels and return the system to the standard system settings, issue the following command:

```
viosecure -level default
```

For more information about using the **viosecure** command, see the [viosecure](#) command description.

Parent topic: [Securing the Virtual I/O Server](#)

Configuring VIOS firewall settings

Enable the Virtual I/O Server (VIOS) firewall to control IP activity.

The VIOS firewall is not enabled by default. To enable the VIOS firewall, you must turn it on by using the [viosecure](#) command with the `-firewall` option. When you enable it, the default setting is activated, which allows access for the following IP services:

- ftp
- ftp-data
- ssh
- web
- https
- rmc
- cimon

Note: The firewall settings are contained in the file `viosecure.ctl` in the `/home/ios/security` directory. If for some reason the `viosecure.ctl` file does not exist when you issue the command to enable the firewall, you receive an error. You can use the `-force` option to enable the standard firewall default ports. For more information about the `force` option see the [viosecure](#) command description.

You can use the default setting or configure the firewall settings to meet the needs of your environment by specifying which ports or port services to allow. You can also turn off the firewall to deactivate the settings.

Use the following tasks at the VIOS command line to configure the VIOS firewall settings:

1. Enable the VIOS firewall by issuing the following command:

```
viosecure -firewall on
```

2. Specify the ports to allow or deny, by using the following command:

```
viosecure -firwall allow | deny -port number
```

3. View the current firewall settings by issuing the following command:

```
viosecure -firewall view
```

4. If you want to disable the firewall configuration, issue the following command:

```
viosecure -firewall off
```

For a full description of any of the viosecure command options, see the [viosecure](#) command description.

Parent topic: [Securing the Virtual I/O Server](#)

Managing the Virtual I/O Server

Find information about managing Virtual I/O Server user types, adding and removing physical resources, and managing logical volumes. Also find information about backing up, restoring, updating, and monitoring the Virtual I/O Server.

Find information about Virtual I/O Server management tasks. Most of the information in this topic is specific to management in an HMC environment. For information about management tasks in an Integrated Virtualization Manager environment, see [Partitioning with the Integrated Virtualization Manager](#).

- **[Managing shared Ethernet adapters](#)**
This topic contains configuration and management information for shared Ethernet adapters, including configuring shared Ethernet adapters, configuring link aggregation, and changing network settings.
- **[Managing Virtual SCSI](#)**
Find instructions for managing virtual storage devices and logical volumes.
- **[Maintaining the Virtual I/O Server](#)**
Find information about updating, backing up, restoring, and monitoring the Virtual I/O Server.

Parent topic: [Using the Virtual I/O Server](#)

Managing shared Ethernet adapters

This topic contains configuration and management information for shared Ethernet adapters, including configuring shared Ethernet adapters, configuring link aggregation, and changing network settings.

- **[Configuring the shared Ethernet adapter](#)**
Find instructions for configuring shared Ethernet adapters.
- **[Configuring a link aggregation device](#)**
Find instructions for configuring a link aggregation.
- **[Changing the network configuration](#)**
Follow these steps to change or remove the network settings on the Virtual I/O Server partition, such as the IP address, subnet mask, gateway, and nameserver address
- **[Network attributes](#)**
Find instructions for managing network attributes.

Configuring the shared Ethernet adapter

Find instructions for configuring shared Ethernet adapters.

To configure the shared Ethernet adapter, follow these steps.

Create the virtual Ethernet adapter for the Virtual I/O Server. Follow these steps:

1. On the HMC, right-click the profile for the Virtual I/O Server and Properties.
2. Create a virtual Ethernet adapter using the Virtual I/O tab by choosing Ethernet in the Create Adapters area.
3. On the Virtual Ethernet Adapter Properties tab, choose the slot number for the virtual adapter and PVID (this PVID will be the default ID used later). Select Trunk Adapter to use this adapter as a gateway between VLANs and an external network. This Ethernet adapter is configured as part of the shared Ethernet adapter.
4. Select the IEEE 802.1Q-compatible adapter check box.
5. If you are using multiple VLANs, add any additional VLAN IDs for the client logical partitions that must communicate with the external network using this virtual adapter.

Repeat this procedure for additional virtual adapters that you require for this partition.

After the virtual Ethernet adapter is created, configure the shared Ethernet adapter. Follow these steps:

1. Verify that the virtual Ethernet trunk adapter is available by typing:

```
lsdev -virtual
```

2. Identify the appropriate physical Ethernet adapter that will be used to create the shared Ethernet adapter on the Virtual I/O Server.

```
lsdev -type adapter
```

You can also use a link aggregation as the shared Ethernet adapter. For configuration information, see [Configuring a link aggregation device](#).

3. Configure the shared Ethernet adapter device by typing the following command:

```
mkvdev -sea target_device -vadapter virtual_ethernet_adapters \
-default DefaultVirtualEthernetAdapter -defaultid SEADefaultPVID
```

The parameters are defined as follows:

- ◆ *target_device* is the physical adapter being used as part of the shared Ethernet adapter device.
- ◆ *virtual_ethernet_adapters* are the virtual Ethernet adapter or adapters that will use the shared Ethernet adapter.
- ◆ *DefaultVirtualEthernetAdapter* is the default virtual Ethernet adapter used to handle untagged packets. If you have only one virtual Ethernet adapter for this partition, use it as the default.
- ◆ *SEADefaultPVID* is the PVID associated with your default virtual Ethernet adapter.

For example, to create shared Ethernet adapter `ent3` with `ent0` as the physical Ethernet adapter (or link aggregation) and `ent2` as the only virtual Ethernet adapter (defined with a PVID of 1), type the following:

```
mkvdev -sea ent0 -vadapter ent2 -default ent2 -defaultid 1
```

4. Verify that the shared Ethernet adapter has been created by typing the following command:

```
lsdev -virtual
```

The shared Ethernet adapter is now configured. After you configure the TCP/IP connections for the virtual adapters on the client logical partitions using the client partitions' operating systems, those partitions can now communicate with the external network.

Additional optional steps

- If you have multiple VLANs using the shared Ethernet adapter, create the VLANs over the shared Ethernet adapter by typing the following command:

```
mkvdev -vlan TargetAdapter -tagid TagID
```

The parameters are defined as follows:

- ◆ *TargetAdapter* is the shared Ethernet adapter.
- ◆ *TagID* is the VLAN ID that you defined when creating the virtual Ethernet adapter associated with the shared Ethernet adapter.

For example, to create a VLAN using the shared Ethernet adapter `ent3` that you just created with a VLAN ID of 1, type the following:

```
mkvdev -vlan ent3 -tagid 1
```

Verify that the VLAN has been created by typing the following command:

```
lsdev -virtual
```

Repeat for any additional VLANs that you defined.

- If you want to be able to access the Virtual I/O Server from the network, for example, using Telnet, configure a TCP/IP connection. If your configuration uses only the PVID and no additional VLANs, configure the interface associated with the shared Ethernet adapter device. If your configuration uses any additional VLANs, configure the interfaces associated with the VLAN devices.

To configure a TCP/IP connection, type the following command:

```
mktcpip -hostname Hostname -inetaddr Address -interface Interface -netmask \
SubnetMask -gateway Gateway -nsrvaddr NameServerAddress -nsrvdomain Domain
```

For example, to configure a TCP/IP connection for the interface associated with the shared Ethernet device that you previously created, use the following parameters:

- ◆ *Hostname* is the host name you defined for this partition
- ◆ *Address* is the IP address you want to associate with the shared Ethernet adapter
- ◆ *Interface* is the interface associated with the shared Ethernet adapter device. In this example, because the shared Ethernet adapter device is `ent3`, the associated interface is `en3`.
- ◆ *Subnetmask* is the subnet mask address that you defined for your subnet.
- ◆ *Gateway* is the gateway address that you defined for your subnet.
- ◆ *NameServerAddress* is the address of your domain name server.
- ◆ *Domain* is the name of your domain.

Tip: If you are using multiple VLANs, use the preceding command to configure only the first TCP/IP connection. For each additional connection, type the following command:

```
chdev -dev interface -perm -attr netaddr=IPaddress -attr netmask=netmask -attr state=up
```

When using this command, enter the interface (`enX`) associated with the device for the *interface* parameter.

Parent topic: [Managing shared Ethernet adapters](#)

Configuring a link aggregation device

Find instructions for configuring a link aggregation.

Configure a link aggregation device by using the `mkvdev` command. A link aggregation device can be used as the physical Ethernet adapter in the shared Ethernet adapter configuration. Configure a link aggregation device by typing the following command:

```
mkvdev -lnagg TargetAdapter ... -attr Attribute=Value ...
```

For example, to create link aggregation device `ent5` with physical Ethernet adapters `ent3`, `ent4`, and backup adapter `ent2`, type the following:

```
mkvdev -lnagg ent3,ent4 -attr backup_adapter=ent2
```

After the link aggregation device is configured, you can add adapters to it, remove adapters from it, or modify its attributes using the `cfglnagg` command.

For more information, see [Link aggregation devices](#).

Parent topic: [Managing shared Ethernet adapters](#)

Changing the network configuration

Follow these steps to change or remove the network settings on the Virtual I/O Server partition, such as the IP address, subnet mask, gateway, and nameserver address

In this scenario, the Virtual I/O Server partition already has its network configuration set. The current configuration will be removed, and the updated configuration will then be set.

1. View the current network configuration using the **lstcpip** command. See [lstcpip Command](#) for details about the **lstcpip** command.
2. Remove the current network configuration by running the **rmtcpip** command. You can remove all network settings or just the specific settings that need to be updated. See [rmtcpip Command](#) for details about the **rmtcpip** command.
3. Configure the new network settings using the **mktcpip** command. See [mktcpip Command](#) for details about the **mktcpip** command.

For example, the Virtual I/O Server partition needs to have its DNS information updated from its current address to 9.41.88.180.

1. Run `lstcpip -namesrv` to view the current configuration. Ensure you want to update this configuration.
2. Run `rmtcpip -namesrv` to remove the current configuration.
3. Run `mktcpip -nsrvaddr 9.41.88.180` to update the nameserver address.

Parent topic: [Managing shared Ethernet adapters](#)

Network attributes

Find instructions for managing network attributes.

Several of the Virtual I/O Server commands, including `chdev`, `mkvdev`, and `cfglnagg`, allow you to change device or network attributes. This section defines attributes that can be modified.

Ethernet Attributes

You can modify the following Ethernet attributes:

Attribute	Description
Maximum Transmission Unit (<i>mtu</i>)	Specifies maximum transmission unit. This value can be any number from 60 through 65535, but it is media dependent.
Interface State (<i>state</i>)	<p>detach Removes an interface from the network interface list. If the last interface is detached, the network interface driver code is unloaded. To change the interface route of an attached interface, that interface must be detached and added again with the <code>chdev -dev <i>Interface</i> -attr <i>state=detach</i></code> command.</p> <p>down Marks an interface as inactive, which keeps the system from trying to transmit messages through that interface. Routes that use the interface, however, are not automatically disabled. (<code>chdev -dev <i>Interface</i> -attr <i>state=down</i></code>)</p> <p>up Marks an interface as active. This parameter is used automatically when setting the first address for an interface. It can also be used to enable an interface after the <code>chdev -dev <i>Interface</i> -attr <i>state=up</i></code> command.</p>
Network Mask (<i>netmask</i>)	<p>Specifies how much of the address to reserve for subdividing networks into subnetworks.</p> <p>The <i>mask</i> includes both the network part of the local address and the subnet part, which is taken from the host field of the address. The mask can be specified as a single hexadecimal number beginning with 0x, in standard Internet dotted-decimal notation.</p> <p>In the 32-bit address, the mask contains bits with a value of 1 for the bit positions reserved for the network and subnet parts, and a bit with the value of 0 for the bit positions that specify the host. The mask contains the standard network portion, and the subnet segment is contiguous with the network segment.</p>

Shared Ethernet adapter attributes

The following shared Ethernet adapter attributes can be modified:

Attribute	Description
PVID (<i>pvid</i>)	Specifies the PVID to use for the shared Ethernet adapter.
PVID adapter (<i>pvid_adapter</i>)	Specifies the default virtual adapter to use for non-VLAN tagged packets.
	Specifies the physical adapter associated with the shared Ethernet adapter.

Physical adapter (<i>real_adapter</i>)	
Thread (<i>thread</i>)	<p>Activates or deactivates threading on the shared Ethernet adapter. Activating this option adds approximately 16% to 20% more overhead for MTU 1500 streaming and 31% to 38% more overhead for MTU 9000. The threading option has more overhead at lower workloads due to the threads being started for each packet. At higher workload rates, such as full duplex or the request/response workloads, the threads can run longer without waiting and being redispached.</p> <p>Threaded mode should be used when Virtual SCSI will be run on the same Virtual I/O Server partition as Shared Ethernet adapter. Threaded mode helps ensure that Virtual SCSI and the Shared Ethernet adapter can share the processor resource appropriately. However, threading adds more instruction path length, which uses additional processor cycles. If the Virtual I/O Server partition will be dedicated to running shared Ethernet devices (and associated virtual Ethernet devices) only, the adapters should be configured with threading disabled.</p> <p>You can enable or disable threading using the <code>-attr thread</code> option of the mkvdev command. To enable threading, use the <code>-attr thread=1</code> option. To disable threading, use the <code>-attr thread=0</code> option. For example, the following command disables threading for shared Ethernet adapter <code>ent1</code>:</p> <pre>mkvdev -sea ent1 -vadapter ent5 -default ent5 -defaultid 1 -attr thread=0</pre>
Virtual adapters (<i>virt_adapter</i>)	Lists the virtual Ethernet adapters associated with the shared Ethernet adapter.
TCP segmentation offload (<i>largesend</i>)	<p>Enables TCP largesend capability (also known as segmentation offload) from logical partitions to the physical adapter. The physical adapter must be enabled for TCP largesend for the segmentation offload from the LPAR to the SEA to work. Also, the LPAR must be capable of performing a largesend operation. On AIX, largesend can be enabled on an LPAR using the ifconfig command.</p> <p>You can enable or disable TCP largesend using the <code>-a largesend</code> option of the chdev command. To enable it, use the <code>'-a largesend=1'</code> option. To disable it, use the <code>'-a largesend=0'</code> option.</p> <p>For example, the following command enables <i>largesend</i> for shared ethernet adapter <code>ent1</code>:</p> <pre>chdev -l ent1 -a largesend=1</pre> <p>By default the setting is disabled (<code>largesend=0</code>).</p>

Shared Ethernet adapter failover attributes

Attribute	Description
High availability mode (<i>ha_mode</i>)	Determines whether the devices participates in a failover setup. The default is <code>disabled</code> . Normally, a shared Ethernet adapter in a failover setup is operating in <code>auto</code> mode, and the primary adapter is decided based on which adapter has the highest priority (lowest numerical value). A shared Ethernet device can be forced into the standby mode, where it will behave as the backup device as long as it can detect the presence of a functional primary.
Control Channel (<i>ctl_chan</i>)	Sets the virtual Ethernet device that is required for a shared Ethernet adapter in a failover setup so that it can communicate with the other adapter. There is no default value for this attribute, and it is required when the ha_mode is not set to <code>disabled</code> .
Internet address to ping (<i>netaddr</i>)	Optional attribute that can be specified for a shared Ethernet adapter that has been configured in a failover setup. When this attribute is specified, a shared

Ethernet device will periodically ping the IP address to verify connectivity (in addition to checking for link status of the physical devices). If it detects a loss of connectivity to the specified ping host, it will initiate a failover to the backup shared Ethernet adapter.

INET attributes

The following INET attributes can be modified:

Attribute	Description
Host Name (<i>hostname</i>)	<p>Specify the host name that you want to assign to the current machine.</p> <p>When specifying the host name, use ASCII characters, preferably alphanumeric only. Do not use a period in the host name. Avoid using hexadecimal or decimal values as the first character (for example <code>3Ccomm</code>, where <code>3C</code> might be interpreted as a hexadecimal character). The unqualified host name should be fewer than 32 characters, for compatibility with earlier hosts.</p> <p>If the host uses a domain name server for name resolution, the host name must contain the full domain name.</p> <p>In the hierarchical domain naming system, names consist of a sequence of subnames that are not case-sensitive and that are separated by periods with no embedded blanks. The DOMAIN protocol specifies that a local domain name must be fewer than 64 characters, and that a host name must be fewer than 32 characters in length. The host name is given first. Optionally, the full domain name can be specified; the host name is followed by a period, a series of local domain names separated by periods, and finally by the root domain. A fully specified domain name for a host, including periods, must be fewer than 255 characters in length and in the following form:</p> <pre>host.subdomain.subdomain.rootdomain</pre> <p>In a hierarchical network, certain hosts are designated as name servers that resolve names into Internet addresses for other hosts. This arrangement has two advantages over the flat name space: resources of each host on the network are not consumed in resolving names, and the person who manages the system does not need to maintain name-resolution files on each machine on the network. The set of names managed by a single name server is known as its <i>zone of authority</i>.</p>
Gateway (<i>gateway</i>)	<p>Identifies the gateway to which packets are addressed. The <i>Gateway</i> parameter can be specified either by symbolic name or numeric address.</p>
Route (<i>route</i>)	<p>Specifies the route. The format of the <i>Route</i> attribute is: <i>route=destination, gateway, metric</i>.</p> <p>destination Identifies the host or network to which you are directing the route. The <i>Destination</i> parameter can be specified either by symbolic name or numeric address.</p> <p>gateway Identifies the gateway to which packets are addressed. The <i>Gateway</i> parameter can be specified either by symbolic name or numeric address.</p> <p>metric Sets the routing metric. The default is 0 (zero). The routing metric is used by the routing protocol (the routed daemon). Higher metrics have the effect of making a route less favorable. Metrics are counted as additional hops to the destination network or host.</p>

Adapter attributes

The following adapter attributes can be modified. The attribute behavior can vary, based on the adapter and driver you have.

Attribute	Adapters/Drivers	Description
Media Speed (<i>media_speed</i>)	<ul style="list-style-type: none"> • 2-Port 10/100/1000 Base-TX PCI-X Adapter • 10/100/1000 Base-T Ethernet PCI-X Adapter Device Driver 	<p>The media speed attribute indicates the speed at which the adapter attempts to operate. The available speeds are 10 Mbps half-duplex, 10 Mbps full-duplex, 100 Mbps half-duplex, 100 Mbps full-duplex and autonegotiation, with a default of autonegotiation. Select auto-negotiate when the adapter should use autonegotiation across the network to determine the speed. When the network will not support autonegotiation, select the specific speed.</p> <p>1000 MBps half and full duplex are not valid values. According to the IEEE 802.3z specification, gigabit speeds of any duplexity must be autonegotiated for copper (TX)-based adapters. If these speeds are desired, select auto-negotiate.</p>
Media Speed (<i>media_speed</i>)	<ul style="list-style-type: none"> • 2-Port Gigabit Ethernet-SX PCI-X Adapter • Gigabit Ethernet-SX PCI-X Adapter Device Driver 	<p>The media speed attribute indicates the speed at which the adapter attempts to operate. The available speeds are 1000 Mbps full-duplex and autonegotiation. The default is autonegotiation. Select auto-negotiate when the adapter should use autonegotiation across the network to determine the duplexity. When the network does not support autonegotiation, select 1000 Mbps full-duplex.</p>
Media Speed (<i>media_speed</i>)	<ul style="list-style-type: none"> • 10/100 Mbps Ethernet PCI Adapter Device Driver 	<p>The media speed attribute indicates the speed at which the adapter attempts to operate. The available speeds are 10 Mbps half-duplex, 10 Mbps full-duplex, 100 Mbps half-duplex, 100 Mbps full-duplex and autonegotiation, with a default of autonegotiation. When the adapter should use autonegotiation across the network to determine the speed, select autonegotiate. When the network will not support autonegotiation, select the specific speed.</p> <p>If autonegotiation is selected, the remote link device must also be set to autonegotiate to ensure the link works correctly.</p>
Media Speed (<i>media_speed</i>)	<ul style="list-style-type: none"> • 10/100/1000 Base-T Ethernet PCI adapter • Gigabit Ethernet-SX PCI Adapter Device Driver 	<p>The media speed attribute indicates the speed at which the adapter attempts to operate. The available speeds are 10 Mbps half-duplex, 10 Mbps full-duplex, 100 Mbps half-duplex, 100 Mbps full-duplex and autonegotiation, with a default of autonegotiation. Select autonegotiate when the adapter should use autonegotiation across the network to determine the speed.</p>

		<p>When the network will not support autonegotiation, select the specific speed.</p> <p>For the adapter to run at 1000 Mbit/s, the autonegotiation setting must be selected.</p> <p>Note: For the Gigabit Ethernet-SX PCI Adapter, the only selection available is autonegotiation.</p>
Enable Alternate Ethernet Address (<i>use_alt_addr</i>)		Setting this attribute to <i>yes</i> indicates that the address of the adapter, as it appears on the network, is the one specified by the Alternate Ethernet Address attribute. If you specify the <i>no</i> value, the unique adapter address written in a ROM on the adapter card is used. The default value is <i>no</i> .
Alternate Ethernet Address (<i>alt_addr</i>)		Allows the adapter unique address, as it appears on the LAN network, to be changed. The value entered must be an Ethernet address of 12 hexadecimal digits and must not be the same as the address of any other Ethernet adapter. There is no default value. This field has no effect unless the Enable Alternate Ethernet Address attribute is set to <i>yes</i> value, in which case this field must be filled in. A typical Ethernet address is 0x02608C000001. All 12 hexadecimal digits, including leading zeros, must be entered.
Enable Link Polling (<i>poll_link</i>)	<ul style="list-style-type: none"> • 10/100Mbps Ethernet PCI Adapter Device Driver 	Select <i>no</i> to cause the device driver to poll the adapter to determine the status of the link at a specified time interval. The time interval value is specified in the Poll Link Time Interval field. If you select <i>no</i> , the device driver will not poll the adapter for its link status. The default value is <i>no</i> .
Poll Link Time Interval (<i>poll_link_time</i>)	<ul style="list-style-type: none"> • 10/100Mbps Ethernet PCI Adapter Device Driver 	The amount of time, in milliseconds, between polls to the adapter for its link status that the device driver is allowed. This value is required when the Enable Link Polling option is set to <i>yes</i> . A value between 100 through 1000 can be specified. The incremental value is 10. The default value is 500.
Flow Control (<i>flow_ctrl</i>)	<ul style="list-style-type: none"> • 10/100/1000 Base-T Ethernet PCI-X Adapter Device Driver • Gigabit Ethernet-SX PCI-X Adapter Device Driver • 2-Port 10/100/1000 Base-TX PCI-X Adapter • 2-Port Gigabit Ethernet-SX PCI-X Adapter • Gigabit Ethernet-SX PCI Adapter Device Driver 	This attribute specifies whether the adapter should enable transmit and receive flow control. The default value is <i>no</i> .
Transmit Jumbo Frames (<i>jumbo_frames</i>)	<ul style="list-style-type: none"> • 10/100/1000 Base-T Ethernet PCI-X Adapter Device Driver 	Setting this attribute to <i>yes</i> indicates that frames up to 9018 bytes in length might be transmitted on this adapter. If you specify <i>no</i> , the maximum size of frames transmitted

	<ul style="list-style-type: none"> • Gigabit Ethernet-SX PCI-X Adapter Device Driver • 2-Port 10/100/1000 Base-TX PCI-X Adapter • 2-Port Gigabit Ethernet-SX PCI-X Adapter • Gigabit Ethernet-SX PCI Adapter Device Driver 	is 1518 bytes. Frames up to 9018 bytes in length can always be received on this adapter.
Checksum Offload (<i>chksum_offload</i>)	<ul style="list-style-type: none"> • 10/100/1000 Base-T Ethernet PCI-X Adapter Device Driver • Gigabit Ethernet-SX PCI-X Adapter Device Driver • 2-Port 10/100/1000 Base-TX PCI-X Adapter • 2-Port Gigabit Ethernet-SX PCI-X Adapter • Gigabit Ethernet-SX PCI Adapter Device Driver • Virtual Ethernet adapters 	<p>Setting this attribute to <code>yes</code> indicates that the adapter calculates the checksum for transmitted and received TCP frames. If you specify <code>no</code>, the checksum will be calculated by the appropriate software.</p> <p>When a virtual Ethernet adapter has checksum offload enabled, the adapter advertises it to the Hypervisor. The Hypervisor tracks which virtual Ethernet adapters have checksum offload enabled and manages inter-partition communication accordingly.</p> <p>When network packets are routed through the shared Ethernet adapter, there is a potential for link errors. In this environment, the packets must traverse the physical link with a checksum. Communication works in the following way:</p> <ul style="list-style-type: none"> • When a packet is received from the physical link, the physical adapter verifies the checksum. If the packet's destination is a virtual Ethernet adapter with checksum offload enabled, the receiver does not have to perform checksum verification. A receiver that does not have checksum offload enabled will accept the packet after checksum verification. • When a packet originates from a virtual Ethernet adapter with checksum offload enabled, it travels to the physical adapter without a checksum. The physical adapter will generate a checksum before sending the packet out. Packets originating from a virtual Ethernet adapter with checksum offload disabled generate the checksum at the source. <p>To enable checksum offload for a shared Ethernet adapter, all constituent devices must have it enabled as well. The shared Ethernet device will fail if the underlying devices do not have the same checksum offload settings.</p>
Enable Hardware Transmit TCP Resegmentation (<i>large_send</i>)	<ul style="list-style-type: none"> • 10/100/1000 Base-T Ethernet PCI-X Adapter Device Driver • Gigabit Ethernet-SX PCI-X Adapter Device Driver • 2-Port 10/100/1000 	This attribute specifies whether the adapter is to perform transmit TCP resegmentation for TCP segments. The default value is <code>no</code> .

	Base-TX PCI-X Adapter • 2-Port Gigabit Ethernet-SX PCI-X Adapter • Gigabit Ethernet-SX PCI Adapter Device Driver	
--	--	--

Link aggregation device attributes

The following link aggregation attributes can be modified:

Attribute	Description
Link aggregation adapters (<i>adapter_names</i>)	The adapters that currently make up the link aggregation device. If you want to modify these adapters, modify this attribute and select all the adapters that should belong to the link aggregation device. When you use this attribute to select all of the adapters that should belong to the link aggregation device, its interface must not have an IP address configured.
Mode (<i>mode</i>)	<p>The type of channel that is configured. In standard mode, the channel sends the packets to the adapter based on an algorithm (the value used for this calculation is determined by the Hash Mode attribute). In round_robin mode, the channel gives one packet to each adapter before repeating the loop. The default mode is standard.</p> <p>The 8023ad mode enables the link aggregation control protocol (LACP) to negotiate the adapters in the link aggregation device with an LACP-enabled switch.</p> <p>If the Hash Mode attribute is set to anything other than default, this attribute must be set to standard or 8023ad. Otherwise, the configuration of the link aggregation device will fail.</p>
Hash Mode (<i>hash_mode</i>)	<p>If operating under standard or IEEE 802.3ad mode, the hash mode attribute determines how the outgoing adapter for each packet is chosen. Following are the different modes:</p> <ul style="list-style-type: none"> • <i>default</i>: uses the destination IP address to determine the outgoing adapter. • <i>src_port</i>: uses the source TCP or UDP port for that connection. • <i>dst_port</i>: uses the destination TCP or UDP port for that connection. • <i>src_dst_port</i>: uses both the source and destination TCP or UDP ports for that connection to determine the outgoing adapter. <p>You cannot use round-robin mode with any hash mode value other than default. The link aggregation device configuration will fail if you attempt this combination.</p> <p>If the packet is not TCP or UDP, it uses the default hashing mode (destination IP address).</p> <p>Using TCP or UDP ports for hashing can make better use of the adapters in the link aggregation device, because connections to the same destination IP address can be sent over different adapters (while still retaining the order of the packets), thus increasing the bandwidth of the link aggregation device.</p>
Internet Address to Ping (<i>netaddr</i>)	This field is optional. The IP address that the link aggregation device should ping to verify that the network is up. This is only valid when there is a backup adapter and when there is just one adapter in the link aggregation device. An address of zero (or all zeros) is ignored and disables the sending of ping packets if a valid address was previously defined. The default is to leave this field blank.
Retry Timeout (<i>retry_time</i>)	This field is optional. It controls how often the link aggregation device sends out a ping packet to poll the current adapter for link status. This is valid only when the

	link aggregation device has only one adapter and a backup adapter defined and the Internet Address to Ping field contains a non-zero address. Specify the timeout value in seconds. The range of valid values is 1 to 100 seconds. The default value is 1 second.
Number of Retries (<i>num_retries</i>)	This field is optional. It specifies the number of lost ping packets before the link aggregation device switches adapters. This is valid only when the channel has one adapter with a backup adapter and the Internet Address to Ping field contains a non-zero address. The range of valid values is 2 to 100 retries. The default value is 3.
Enable Gigabit Ethernet Jumbo Frames (<i>use_jumbo_frame</i>)	This field is optional. To use this attribute, your switch must support jumbo frames. This will work only with a Standard Ethernet (en) interface, not an IEEE 802.3 (et) interface.
Enable Alternate Address (<i>use_alt_addr</i>)	This field is optional. Setting this to yes will enable you to specify a MAC address that you want the link aggregation device to use. If you set this option to no, the link aggregation device will use the MAC address of the first adapter.
Alternate Address (<i>alt_addr</i>)	If Enable Alternate Address is set to <code>yes</code> , specify the MAC address that you want to use. The address you specify must start with <code>0x</code> and be a 12-digit hexadecimal address.

VLAN attributes

The following VLAN attributes can be modified:

Attribute	Value
VLAN Tag ID (<i>vlan_tag_id</i>)	The unique ID associated with the VLAN driver. You can specify from 1 to 4094.
Base Adapter (<i>base_adapter</i>)	The network adapter to which the VLAN device driver is connected.

Parent topic: [Managing shared Ethernet adapters](#)

Managing Virtual SCSI

Find instructions for managing virtual storage devices and logical volumes.

Provisioning virtual disk resources occurs on the Virtual I/O Server. Physical disks owned by the Virtual I/O Server can either be exported and assigned to a client partition as a whole or can be partitioned into logical volumes. These logical volumes can be exported as virtual disks to one or more client partitions. Therefore, Virtual SCSI enables sharing of adapters and disk devices.

To make a physical or logical volume available to a client partition requires that it be assigned to a Virtual SCSI server adapter on the Virtual I/O Server. The SCSI client adapter is linked to a particular virtual SCSI server adapter in the Virtual I/O Server partition. The client partition accesses its assigned disks through the Virtual SCSI client adapter. The Virtual I/O Server client adapter sees standard SCSI devices and LUNs through this virtual adapter. Assigning disk resources to a SCSI server adapter in the Virtual I/O Server effectively allocates resources to a SCSI client adapter in the client partition.

- **Identifying exportable disks**

In order to export a physical volume as a virtual device, the physical volume must have either a unique identifier (UDID), a physical identifier (PVID), or an IEEE volume attribute. Use this procedure to identify which disks have an identifier.

- **Creating the virtual target device on the Virtual I/O Server**
Find instructions for creating a virtual target device on the Virtual I/O Server.
- **Creating volume groups and logical volumes on the Virtual I/O Server**
Find instructions for creating logical volumes and volume groups on the Virtual I/O Server.
- **Importing or Exporting a Volume Group**
Find instructions for importing and exporting volume groups.
- **Mapping virtual disks to physical disks**
Find instructions for mapping a virtual disk on a client logical partition to its physical disk on the Virtual I/O Server.
- **Increasing Virtual SCSI device capacity**
Increase the size of Virtual SCSI disks.
- **Changing the Virtual SCSI queue depth**
Increasing the Virtual SCSI queue depth might provide performance improvements for some virtual configurations. Understand the factors involved in determining a change to the Virtual SCSI queue depth value.
- **Virtual SCSI reserve/release requirements**
Understand the Virtual SCSI setup requirements to support applications using SCSI reserve and release.

Parent topic: [Managing the Virtual I/O Server](#)

Identifying exportable disks

In order to export a physical volume as a virtual device, the physical volume must have either a unique identifier (UDID), a physical identifier (PVID), or an IEEE volume attribute. Use this procedure to identify which disks have an identifier.

Parent topic: [Managing Virtual SCSI](#)

Listing disks with a UDID

Follow these steps to list disks that have a UDID:

1. From the Virtual I/O Server command line, type `oem_setup_env`.
2. Type `odmget -qattribute=unique_id CuAt`. The disks that have a UDID are listed. Output similar to the following is displayed:

```
CuAt:
  name = "hdisk1"
  attribute = "unique_id"
  value = "2708ECVBZ1SC10IC35L146UCDY10-003IBMscsi"
  type = "R"
  generic = ""
  rep = "nl"
  nls_index = 79

CuAt:
  name = "hdisk2"
  attribute = "unique_id"
  value = "210800038FB50AST373453LC03IBMscsi"
  type = "R"
  generic = ""
  rep = "nl"
  nls_index = 79
```

Devices in the list that are accessible from other Virtual I/O Server partitions can be used in Virtual SCSI MPIO configurations.

3. Type `exit`.

Listing disks with a PVID

Follow these steps to list disks that have a PVID and, if necessary, add a PVID to a disk.

1. To list devices with a PVID, type `lspv`. If the second column has a value of `none`, the physical volume does not have a PVID. A PVID must be put on the physical volume before it can be exported as a virtual device.
2. To do this, run the following command from the Virtual I/O Server command-line interface: `chdev -dev physicalvolumename -attr pv=yes -perm`

Listing disks with an IEEE volume attribute

To determine whether a device has an IEEE volume attribute identifier, type the following command:

```
lsattr -l hdiskX
```

Creating the virtual target device on the Virtual I/O Server

Find instructions for creating a virtual target device on the Virtual I/O Server.

The following procedure describes how to configure Virtual SCSI. This procedure can be repeated to provide additional virtual disk storage to any client logical partition. This procedure assumes that you already have a physical or logical volume defined on the Virtual I/O Server. For information about physical and logical volumes, see [Logical volumes](#).

This procedure also assumes that the virtual adapters for the Virtual I/O Server and the client partitions were created during the creation of the partition profile. For information about creating the partition, see [Creating the Virtual I/O Server logical partition and partition profile](#).

With the Virtual I/O Server you can export disks as virtual disks. With the Virtual I/O Server you can export two types of physical disks: Virtual SCSI disk backed by a physical volume and a Virtual SCSI disk backed by a logical volume. After a virtual disk is assigned to a client partition, the Virtual I/O Server must be available before the client logical partitions can access it.

Creating the virtual target device on the Virtual I/O Server maps the Virtual SCSI adapter with the logical volume or physical disk. This can be accomplished with the `mkvdev` command. The syntax for this command is as follows:

```
mkvdev -vdev TargetDevice -vadapter VirtualSCSIServerAdapter -dev DeviceName
```

1. Use the `lsdev` command to ensure that the virtual SCSI adapter is available. For example, running `lsdev -virtual` returns results similar to the following:

```
name status description
ent2 Available Virtual I/O Ethernet Adapter (1-lan)
vhost0 Available Virtual SCSI Server Adapter
vhost1 Available Virtual SCSI Server Adapter
vhost2 Available Virtual SCSI Server Adapter
vsa0 Available LPAR Virtual Serial Adapter
```

2. To create a virtual target device, which maps the virtual SCSI server adapter to a physical or logical volume, run the `mkvdev` command. In this procedure, we ran the following command

```
mkvdev vdev lv_4G vadapter vhost3
```

In this example, the name of the virtual SCSI server adapter is `vhost3`. The logical volume that was specified was `lv_4G`.

Note: The `-vdev` flag can specify either a physical or logical volume or an optical device. To map a physical volume to the virtual SCSI server adapter, use `hdiskx` for the `-vdev` flag. For example, if the physical volume name was `hdisk5`, run `mkvdev -vdev hdisk5 -vadapter vhost3`. To map an optical device to the virtual SCSI server adapter, use `cdx` for the `-vdev` flag. For example, if the optical device name is `cd0`, run `mkvdev -vdev cd0 -vadapter vhost3`.

The storage is available to the client partition either the next time it starts, or the next time the appropriate virtual SCSI client adapter is probed (on a Linux logical partition), or configured (on an AIX logical partition).

To map a physical volume to the Virtual SCSI Server Adapter, use `hdiskx` instead of the logical volume devices for the `-vdev` flag.

The `lsdev` command shows the newly created Virtual Target Device adapter. For example, running `lsdev -virtual` returns results similar to the following:

```
name status description
vhost0 Available Virtual SCSI Server Adapter
vsa0 Available LPAR Virtual Serial Adapter
vdbsrv Available Virtual Target Device - Logical Volume
```

The `lsmap` command shows the logical connections between newly created devices, as follows:

```
lsmap -vadapter vhost0
```

This command returns results similar to the following:

```
SVSA Physloc Client PartitionID
-----
vhost0 U9111.520.10DDEEC-V1-C20 0x00000000
VTD vdbsrv
LUN 0x8100000000000000
Backing device rootvg_dbsrv
Physloc
```

The physical location is a combination of the slot number, in this case 20, and the logical partition ID. The virtual device can now be attached from the client partition. You can now activate your partition into the SMS menus and install the operating system on the virtual disk or add an additional virtual disk by using the `cfgmgr` command. The Client Partition ID is displayed as soon the client partition is active.

For a detailed description of each Virtual I/O Server command, see [Virtual I/O Server command descriptions](#).

Parent topic: [Managing Virtual SCSI](#)

Creating volume groups and logical volumes on the Virtual I/O Server

Find instructions for creating logical volumes and volume groups on the Virtual I/O Server.

To create a logical volume, use the `mklv` command. To create the logical volume on a separate disk, you must first create a volume group and assign one or more disks by using the `mkvg` command. For conceptual information about logical volumes, see [Concepts for Virtual SCSI](#).

1. Create a volume group and assign a disk to this volume group by using the `mkvg` command. In this example, the name of the volume group is `rootvg_clients`

```
mkvg -f -vg rootvg_clients hdisk2 rootvg_clients
```

2. Define the logical volume, which will be visible as a disk to the client partition. The size of this logical volume will act as the size of disks that will be available to the client partition. Use the `mklv` command to create a 2 GB logical volume called `rootvg_dbsrv` as follows:

```
mklv -lv rootvg_dbsrv rootvg_clients 2G rootvg_dbsrv
```

For a detailed description of each Virtual I/O Server command, see [Virtual I/O Server command descriptions](#).

Parent topic: [Managing Virtual SCSI](#)

Importing or Exporting a Volume Group

Find instructions for importing and exporting volume groups.

The following procedure explains how to use the import and export procedures to move a user-defined volume group from one system to another. (The `rootvg` volume group cannot be exported or imported.) The export procedure removes the definition of a volume group from a system. The import procedure introduces the volume group to its new system. You can also use the import procedure to reintroduce a volume group to the system that had been previously associated with and had been exported from. You can also use import and export to add a physical volume that contains data to a volume group by putting the disk to be added in its own volume group.

Note: The `importvg` command changes the name of an imported logical volume if a logical volume of that name already exists on the new system. If the `importvg` command must rename a logical volume, it prints an error message to standard error.

To export a volume group, type the following commands:

1. `deactivatevg VolumeGroupName`
2. `exportvg VolumeGroupName`

To import a volume group, use the `importvg` command.

Parent topic: [Managing Virtual SCSI](#)

Mapping virtual disks to physical disks

Find instructions for mapping a virtual disk on a client logical partition to its physical disk on the Virtual I/O Server.

This procedure shows how to map a Virtual SCSI disk on an AIX client logical partition to the physical device (disk or logical volume) on the Virtual I/O Server.

To map a virtual disk to a physical disk, you need the following information. This information is gathered during this procedure:

- Virtual device name
- Slot number of the Virtual SCSI client adapter
- Logical unit number (LUN) of the Virtual SCSI device
- Client partition ID

Follow these steps to map a virtual disk on an AIX client logical partition to its physical disk on the Virtual I/O Server:

1. Display Virtual SCSI device information on the AIX client logical partition by typing the following command:

```
lscfg -l devicename
```

This command returns results similar to the following:

```
U9117.570.1012A9F-V3-C2-T1-L810000000000 Virtual SCSI Disk Drive
```

2. Record the slot number, which is located in the output, following the card location label *C*. This identifies the slot number of the Virtual SCSI client adapter. In this example, the slot number is 2.
3. Record the LUN, which is located in the output, following the LUN label *L*. In this example, the LUN is 810000000000.
4. Record the partition ID of the client partition. On the AIX client logical partition, type the following command from the Virtual I/O Server command line:
 - a. `oem_setup_env`.
 - b. Run `uname -L`

Your results should look similar to the following:

```
2 fumi02
```

The partition ID is the first number listed. In this example, the partition ID is 2. This number is used in the next step.

- c. Type `exit`.
5. If you have multiple Virtual I/O Server partitions running on your system, determine which Virtual I/O Server partition is serving the Virtual SCSI device. Use the slot number of the client adapter that is linked to a Virtual I/O Server, and a server adapter. Use the HMC command line to list information about Virtual SCSI client adapters in the client logical partition.

Log in to the HMC, and from the HMC command line, type `lshwres`. Specify the managed console name for the `-m` parameter and the client partition ID for the `lpar_ids` parameter.

Note:

- ◆ The managed console name, which is used for the `-m` parameter, is determined by typing `lssyscfg -r sys -F name` from the HMC command line.
- ◆ Use the client partition ID recorded in Step 4 for the `-lpar_ids` parameter.

For example:

```
lshwres -r virtualio --rsubtype scsi -m fumi --filter lpar_ids=2
```

This example returns results similar to the following:

```
lpar_name=fumi02,lpar_id=2,slot_num=2,state=null,adapter_type=client,remote_lpar_id=1,remote_lpar_name=fumi01,remote_slot_num=2,is_required=1,backing_devices=none
```

Record the name of the Virtual I/O Server located in the `remote_lpar_name` field and slot number of the Virtual SCSI server adapter, which is located in the `remote_lpar_id` field. In this example, the

- name of the Virtual I/O Server is fumi01 and the slot number of the Virtual SCSI server adapter is 1.
6. Log in to the Virtual I/O Server.
 7. List virtual adapters and devices on the Virtual I/O Server by typing the following command:

```
lsmmap -all
```

8. Find the Virtual SCSI server adapter (vhostX) that has a slot ID that matches the remote slot ID recorded in Step 7. On that adapter, run the following command:

```
lsmmap -vadapter devicename
```

9. From the list of devices, match the LUN recorded in Step 4 with LUNs listed. This is the physical device.

Parent topic: [Managing Virtual SCSI](#)

Increasing Virtual SCSI device capacity

Increase the size of Virtual SCSI disks.

As storage demands increase for virtual client partitions, you can add physical storage to increase the size of your virtual devices and allocate that storage to your virtual environment, without disrupting client operations. You can increase the capacity of your Virtual SCSI devices by adding physical volumes to volume groups. To add physical volumes to a volume group, use the **extendvg** command. For more information about using this command, see the [extendvg](#) command description.

After you increase the size of a volume group, you can allocate the increased volume to logical partitions by resizing logical volumes. To increase the size of a logical volume, use the **extendlv** command. For information about using this command, see the [extendlv](#) command description.

Shutting down or reconfiguring a partition is not required to begin using the additional resources. If the physical storage resources have been set up and properly allocated to the system as a system resource, as soon as the Virtual I/O Server recognizes the changes in storage volume the increased storage capacity is available to the Virtual I/O clients.

Parent topic: [Managing Virtual SCSI](#)

Changing the Virtual SCSI queue depth

Increasing the Virtual SCSI queue depth might provide performance improvements for some virtual configurations. Understand the factors involved in determining a change to the Virtual SCSI queue depth value.

The Virtual SCSI queue depth value determines how many requests the disk head driver will queue to the Virtual SCSI client driver at any one time. You can change this value from the default value to any value from 1 to 256. The default value is 3. You modify this value using the `chdev` command. For more information about this command, see [chdev command](#).

Increasing this value might improve the throughput of the disk in specific configurations. However, several factors must be taken into consideration. These factors include the value of the queue-depth attribute for all of the physical storage devices on the Virtual I/O Server being used as a virtual target device by the disk instance on the client partition, and the maximum transfer size for the virtual SCSI adapter instance that is the parent device for the disk instance.

The maximum transfer size for Virtual SCSI client adapters is set by the Virtual I/O Server, which determines the value based on the resources available on the server and the maximum transfer size set for the physical storage devices on that server. Other factors include the queue depth and maximum transfer size of other devices involved in mirrored-volume-group or Multipath I/O (MPIO) configurations. Increasing the queue depth for some devices might reduce the resources available for other devices on that same shared adapter and

decrease the throughput for those devices.

To change the queue depth, on the client partition use the `chdev` command with the `queue_depth=value` attribute as in the following example:

```
chdev -l hdiskN -a "queue_depth=value"
```

hdiskN represents the name of a physical volume and *value* is the value you assign between 1 and 256.

To view the current setting for the `queue_depth` value, from the client partition issue the following command:

```
lsattr -El hdiskN
```

Parent topic: [Managing Virtual SCSI](#)

Virtual SCSI reserve/release requirements

Understand the Virtual SCSI setup requirements to support applications using SCSI reserve and release.

Virtual I/O Server versions 1.3 and later provide support for applications that are enabled to use SCSI-2 reserve functions that are controlled by the client partition. Typically, SCSI reserve/release is used in clustered environments where contention for SCSI disk resources might require greater control. To ensure Virtual I/O Server support of these environments, configure the VIOS enablement of this support. If the applications you are using provide information about the policy to use for enabling the SCSI-2 reserve functions on the client partition, follow those procedures for setting the reserve policy.

Complete the following tasks to enable the Virtual I/O Server support of SCSI-2 reserve environments:

1. Configure the Virtual I/O Server `reserve_policy` for `single_path`, using the following command:

```
chdev -devl hdiskN -attr reserve_policy=single_path
```

Note: Perform this task when the device is not in use. If you run this command while the device is open or in use, then you must use the `-perm` flag with this command. If you use the `-perm` flag, the changes do not take effect until the device is unconfigured and reconfigured.

2. Enable the `client_reserve` feature on the Virtual I/O Server.
 - ◆ If you are creating a virtual target device, use the following command:

```
mkvdev -vdev hdiskN -vadapter vhostN -attr client_reserve=yes
```

where *hdiskN* is the virtual target device name and *vhostN* is the Virtual SCSI server adapter name.

- ◆ If the virtual target device has already been created, use the following command:

```
chdev -dev vtscsiN -attr client_reserve=yes
```

where *vtscsiN* is the virtual device name.

3. On the Virtual client, complete the following steps to configure the SCSI reserve/release support for the virtual disk backed by the physical disk that you configured in step 1:
 - a. Set the reserve policy on the Virtual client to `single_path`, using the following command:

```
chdev -a reserve_policy=single_path -l hdiskN
```

where *hdiskN* is the virtual disk name

Note: Perform this task when the device is not in use. If you run this command while the device is open or in use, then you must use the `-p` flag. In that case, the changes do not take effect until the device is unconfigured and reconfigured.

- b. Set the `hcheck_cmd` attribute so that the MPIO code uses the inquiry option. If the `hcheck_cmd` attribute is set to **test unit ready** and the backing device is reserved, then `test unit ready` will fail and log an error on the client.

```
chdev -a hcheck_cmd=inquiry -l hdiskN
```

where *hdiskN* is the virtual disk name.

For more information about these commands, see the [Virtual I/O Server command descriptions](#)

Parent topic: [Managing Virtual SCSI](#)

Maintaining the Virtual I/O Server

Find information about updating, backing up, restoring, and monitoring the Virtual I/O Server.

- **[Backing up and restoring the Virtual I/O Server](#)**
Find information, including limitations, about backing up and restoring the Virtual I/O Server. Follow these procedures to back up and restore the Virtual I/O Server.
- **[Monitoring the Virtual I/O Server](#)**
Find information about monitoring the Virtual I/O Server.
- **[Managing user types on the Virtual I/O Server](#)**
Find instructions for creating the system administrator, service representative user, and development engineer user ID.

Parent topic: [Managing the Virtual I/O Server](#)

Backing up and restoring the Virtual I/O Server

Find information, including limitations, about backing up and restoring the Virtual I/O Server. Follow these procedures to back up and restore the Virtual I/O Server.

For information about backing up and restoring the Virtual I/O Server in an Integrated Virtualization Manager environment, see [Backing up and restoring partition data](#).

Attention: Before backing up and restoring the Virtual I/O Server using the procedures described in the following topics, be aware of the limitations involving backing up and restoring logical volumes that are part of the root volume group (rootvg). These limitations are discussed in the following topics.

- **[Backing up the Virtual I/O Server](#)**
This topic contains instructions for backing up the Virtual I/O Server to CD or DVD, tape, or a remote file system.
- **[Restoring the Virtual I/O Server](#)**
This topic contains instructions for restoring the Virtual I/O Server from a CD or DVD, tape, or remote file system.

Parent topic: [Maintaining the Virtual I/O Server](#)

Backing up the Virtual I/O Server

This topic contains instructions for backing up the Virtual I/O Server to CD or DVD, tape, or a remote file system.

Use the `backupios` command to back up the Virtual I/O Server. The backup that is created is a bootable image that will install from the device specified when you created the backup. Backups sent to a file are reinstalled from the HMC using the `installios` command. The `backupios` command backs up the Virtual I/O Server; it does not back up any data stored in user-defined volume groups or logical volumes.

Attention: It is best to avoid using logical volumes that are part of the root volume group (`rootvg`) when creating virtual target devices because these virtual devices will not be made available when the Virtual I/O Server is restored after a failure. If the Virtual I/O Server partition has only one physical volume, `rootvg`, you must recreate these virtual devices after a restore. To get configuration information about virtual devices, use the [lsmap command](#).

Note: In an Integrated Virtualization Manager environment, use the `bkprofdata` command or the Integrated Virtualization Manager interface to backup your profile data before running the `backupios` command.

Back up the Virtual I/O Server each time that a configuration change is made. Adding, deleting, or changing device configurations results in changes to customized data. This customized data is required to be able to re-create the virtual I/O environment in case of unrecoverable failure or platform migration.

- [Backing up the Virtual I/O Server to tape](#)
This topic contains instructions for backing up the Virtual I/O Server to tape.
- [Backing up the Virtual I/O Server to a remote file system](#)
This topic contains instructions for backing up the Virtual I/O Server to a remote file system.
- [Backing up the Virtual I/O Server to DVD](#)
This topic contains instructions for backing up the Virtual I/O Server to DVD.

Parent topic: [Backing up and restoring the Virtual I/O Server](#)

Backing up the Virtual I/O Server to tape

This topic contains instructions for backing up the Virtual I/O Server to tape.

To create a backup to tape, follow these steps:

1. Assign a tape drive to the Virtual I/O Server.
2. Get the device name by typing the following command:

```
lsdev -type tape
```

If the tape device is in the `Defined` state, type the following command, where `dev` is the name of your tape device:

```
cfgdev -dev dev
```

3. Type the following command, where `tape_device` is the name of the tape device you want to back up to:

```
backupios -tape tape_device
```

This command creates a bootable tape that you can use to restore the Virtual I/O Server.

Parent topic: [Backing up the Virtual I/O Server](#)

Backing up the Virtual I/O Server to a remote file system

This topic contains instructions for backing up the Virtual I/O Server to a remote file system.

Backing the Virtual I/O Server up to a file system requires the file system to be available and mounted. Backing up the Virtual I/O Server to a remote file system will create the `nim_resources.tar` image in the directory you specify. The Virtual I/O Server must have root write access to the server on which the backup will be created.

To back the Virtual I/O Server up to a remote file system, follow these steps:

1. Create a mount directory where the backup image, `nim_resources.tar`, will be written. For example, to create the directory `/home/backup`, type:

```
mkdir /home/backup
```

2. Mount an exported directory on the mount directory. For example:

```
mount server1:/export/mksysb_ios /home/backup
```

3. Run the `backupios` command with the `-file` option. Specify the path to the mounted directory. For example:

```
backupios -file /home/backup
```

Parent topic: [Backing up the Virtual I/O Server](#)

Backing up the Virtual I/O Server to DVD

This topic contains instructions for backing up the Virtual I/O Server to DVD.

To back up the Virtual I/O Server to a DVD-RAM disc, follow these steps. Only DVD-RAM media can be used to back up the Virtual I/O Server.

Note: Vendor disc drives may support burning to additional disc types, such as CD-RW and DVD-R. Refer to the documentation for your drive to determine which disc types are supported.

1. Assign an optical drive to the Virtual I/O Server partition.
2. Get the device name by typing the following command:


```
lsdev -type optical
```

If the device is in the `Defined` state, type:

```
cfgdev -dev dev
```

3. Run the **backupios** command with the `-cd` option. Specify the path to the device. For example:

```
backupios -cd /dev/cd0
```

Parent topic: [Backing up the Virtual I/O Server](#)

Restoring the Virtual I/O Server

This topic contains instructions for restoring the Virtual I/O Server from a CD or DVD, tape, or remote file system.

Attention: Virtual target devices created from logical volumes that are part of the root volume group (rootvg) will not be made available when the Virtual I/O Server is restored after a failure. If the Virtual I/O Server partition has only one physical volume, rootvg, you must remove the virtual target devices after the restore and then recreate them using the same logical volume names and virtual server adapters used initially to configure the virtual devices. To get the virtual device information and determine which devices need to be removed, use the [lsmap](#) command. After recreating the virtual devices, the client partitions might indicate that data has been copied to a new physical volume.

For information about backing up the Virtual I/O Server, see [Backing up the Virtual I/O Server](#).

- [Restoring the Virtual I/O Server from tape](#)
This topic contains instructions for restoring the Virtual I/O Server from tape.
- [Restoring the Virtual I/O Server from a remote file system](#)
Restore the Virtual I/O Server from a backup image stored in a remote file system.
- [Restoring the Virtual I/O Server from a CD or DVD](#)
This topic contains instructions for restoring the Virtual I/O Server from a CD or DVD.

Parent topic: [Backing up and restoring the Virtual I/O Server](#)

Restoring the Virtual I/O Server from tape

This topic contains instructions for restoring the Virtual I/O Server from tape.

To restore the Virtual I/O Server from tape, follow these steps:

1. Specify the Virtual I/O Server partition to boot from the tape by using the `bootlist` command. Alternatively, you can alter the bootlist in the System Management Services (SMS).
2. Follow the remaining installation steps according to the system prompts.

Parent topic: [Restoring the Virtual I/O Server](#)

Restoring the Virtual I/O Server from a remote file system

Restore the Virtual I/O Server from a backup image stored in a remote file system.

To restore the Virtual I/O Server from a backup image in a file system, run the **installios** command from the HMC command line. This procedure restores a backup image that was created using the **backupios** command. For more information about **backupios**, see [backupios command](#).

Follow the installation procedures according to the system prompts to specify the backup image to install. For additional information about the **installios** command options, see [installios command](#) in the [Virtual I/O Server Commands Reference](#).

Parent topic: [Restoring the Virtual I/O Server](#)

Restoring the Virtual I/O Server from a CD or DVD

This topic contains instructions for restoring the Virtual I/O Server from a CD or DVD.

To restore the Virtual I/O Server from a CD or DVD, follow these steps:

1. Specify the Virtual I/O Server partition to boot from the CD or DVD by using the bootlist command. Alternatively, you can alter the bootlist in the System Management Services (SMS).
2. Follow the remaining installation steps according to the system prompts.

Parent topic: [Restoring the Virtual I/O Server](#)

Monitoring the Virtual I/O Server

Find information about monitoring the Virtual I/O Server.

AIX and Linux client logical partitions log errors against failing I/O operations. Hardware errors on the client logical partitions associated with virtual devices usually have corresponding errors logged on the server. However, if the failure is within the client partition, there will not be errors on the server. Also, on Linux client logical partitions, if the algorithm for retrying SCSI temporary errors is different from the algorithm used by AIX, the errors might not be recorded on the server.

Parent topic: [Maintaining the Virtual I/O Server](#)

Managing user types on the Virtual I/O Server

Find instructions for creating the system administrator, service representative user, and development engineer user ID.

When the Virtual I/O Server is installed, the only user type that is active is the prime administrator (padmin). For information about creating other user types on the Virtual I/O Server, see [User types](#).

Parent topic: [Maintaining the Virtual I/O Server](#)

Troubleshooting the Virtual I/O Server

Find information about diagnosing Virtual I/O Server problems and information about how to correct those problems.

This section includes information about troubleshooting the Virtual I/O Server. For information about troubleshooting the Integrated Virtualization Manager, see [Troubleshooting with the Integrated Virtualization Manager](#).

- **[Troubleshooting the Virtual I/O Server logical partition](#)**
Find information and procedures for troubleshooting and diagnosing the Virtual I/O Server partition.
- **[Troubleshooting the client logical partition](#)**
Find information and procedures for troubleshooting the client partitions.

Parent topic: [Using the Virtual I/O Server](#)

Troubleshooting the Virtual I/O Server logical partition

Find information and procedures for troubleshooting and diagnosing the Virtual I/O Server partition.


- **[Troubleshooting Virtual SCSI problems](#)**
Find information and procedures for troubleshooting Virtual SCSI problems in the Virtual I/O Server.
- **[Troubleshooting network problems](#)**
Find information and procedures for troubleshooting virtual Ethernet and shared Ethernet adapter problems.

Parent topic: [Troubleshooting the Virtual I/O Server](#)

Troubleshooting Virtual SCSI problems

Find information and procedures for troubleshooting Virtual SCSI problems in the Virtual I/O Server.

For problem determination and maintenance, use the **diagmenu** command provided by the Virtual I/O Server.

Refer to the [AIX fast-path problem-isolation](#) documentation  in the Service provider information because, in certain cases, the diagnostic procedures described in the AIX fast-path problem-isolation documentation are not available from the **diagmenu** command menu. In those cases, run the `oem_setup_env` command on the Virtual I/O Server, then continue with the procedure.

For a detailed description of each Virtual I/O Server command, see [Virtual I/O Server command descriptions](#).

Parent topic: [Troubleshooting the Virtual I/O Server logical partition](#)

Troubleshooting network problems

Find information and procedures for troubleshooting virtual Ethernet and shared Ethernet adapter problems.

Use the `entstat` command to troubleshoot problems. The `entstat` command output for a shared Ethernet adapter has the following sections:

- Shared Ethernet statistics summary
 - ◆ Device information
 - ◆ Sum of the associated physical and virtual adapter statistics
- Statistics that the Shared Ethernet adapter records
 - ◆ Number of adapters: number of real and virtual adapters
 - ◆ VLAN IDs: The VLAN IDs that shared Ethernet adapter learned over time while the packets were received
 - ◆ Shared Ethernet adapter flags
 - ◆ Physical, virtual, and other statistics
- Listing of the individual adapter statistics for the adapters associated with the shared Ethernet adapter

For example, running the `entstat` command returns results similar to the following.

```
entstat    all ent3

ETHERNET STATISTICS (ent3) :
Device Type: Shared Ethernet Adapter
Hardware Address: 00:06:29:6b:18:f4
Elapsed Time: 0 days 0 hours 0 minutes 0 seconds

Transmit Statistics:
-----
Packets: 46
Bytes: 2944
Interrupts: 43
Transmit Errors: 0
Packets Dropped: 0

Receive Statistics:
-----
Packets: 46
Bytes: 2944
Interrupts: 47
Receive Errors: 0
Packets Dropped: 0
Bad Packets: 0

Max Packets on S/W Transmit Queue: 3
S/W Transmit Queue Overflow: 0
Current S/W+H/W Transmit Queue Length: 1

Broadcast Packets: 44
Multicast Packets: 2
No Carrier Sense: 0
DMA Underrun: 0
Lost CTS Errors: 0
Max Collision Errors: 0
Late Collision Errors: 0
Deferred: 0
SQE Test: 0
Timeout Errors: 0
Single Collision Count: 0
Multiple Collision Count: 0
Current HW Transmit Queue Length: 1

Broadcast Packets: 44
Multicast Packets: 2
CRC Errors: 0
DMA Overrun: 0
Alignment Errors: 0
No Resource Errors: 0
Receive Collision Errors: 0
Packet Too Short Errors: 0
Packet Too Long Errors: 0
Packets Discarded by Adapter: 0
Receiver Start Count: 0

General Statistics:
-----
No mbuf Errors: 0
Adapter Reset Count: 0
Driver Flags: Up Broadcast Running
              Simplex 64BitSupport

-----
Statistics for adapters in the Shared Ethernet Adapter ent3
-----
Number of adapters: 2
SEA Flags: 00000000
VLAN IDs :
  ent2: 1 10
Real Side Statistics:
  Packets received: 0
  Packets bridged: 0
  Packets consumed: 0
  Packets fragmented: 0
  Packets transmitted: 0
  Packets dropped: 0
Virtual Side Statistics:
```

Using the Virtual I/O Server

Packets received: 47
Packets bridged: 47
Packets consumed: 47
Packets fragmented: 0
Packets transmitted: 47
Packets dropped: 0
Other Statistics:
Output packets generated: 47
Output packets dropped: 0
Device output failures: 94
Memory allocation failures: 0
ICMP error packets sent: 0
Non IP packets larger than MTU: 0

Real Adapter: ent1

ETHERNET STATISTICS (ent1) :
Device Type: 10/100/1000 Base-T Ethernet PCI Adapter (14100401)
Hardware Address: 00:06:29:6b:18:f4

Transmit Statistics: -----	Receive Statistics: -----
Packets: 0	Packets: 0
Bytes: 0	Bytes: 0
Interrupts: 0	Interrupts: 0
Transmit Errors: 0	Receive Errors: 0
Packets Dropped: 0	Packets Dropped: 0
	Bad Packets: 0

Max Packets on S/W Transmit Queue: 0
S/W Transmit Queue Overflow: 0
Current S/W+H/W Transmit Queue Length: 0

Broadcast Packets: 0	Broadcast Packets: 0
Multicast Packets: 0	Multicast Packets: 0
No Carrier Sense: 0	CRC Errors: 0
DMA Underrun: 0	DMA Overrun: 0
Lost CTS Errors: 0	Alignment Errors: 0
Max Collision Errors: 0	No Resource Errors: 0
Late Collision Errors: 0	Receive Collision Errors: 0
Deferred: 0	Packet Too Short Errors: 0
SQE Test: 0	Packet Too Long Errors: 0
Timeout Errors: 0	Packets Discarded by Adapter: 0
Single Collision Count: 0	Receiver Start Count: 0
Multiple Collision Count: 0	
Current HW Transmit Queue Length: 0	

General Statistics:

No mbuf Errors: 0
Adapter Reset Count: 0
Adapter Data Rate: 2000
Driver Flags: Up Broadcast Simplex
Limbo Promiscuous AlternateAddress
64BitSupport PrivateSegment DataRateSet

Adapter Specific Statistics:

Additional Driver Flags: Autonegotiate
Entries to transmit timeout routine: 0
Firmware Level: 13.0.10
Transmit and Receive Flow Control Status: Disabled
Link Status: Down
Media Speed Selected: Autonegotiation
Media Speed Running: Unknown
Packets with Transmit collisions:
1 collisions: 0 6 collisions: 0 11 collisions: 0
2 collisions: 0 7 collisions: 0 12 collisions: 0
3 collisions: 0 8 collisions: 0 13 collisions: 0
4 collisions: 0 9 collisions: 0 14 collisions: 0
5 collisions: 0 10 collisions: 0 15 collisions: 0

Virtual Adapter: ent2

ETHERNET STATISTICS (ent2) :
Device Type: Gigabit Ethernet-SX PCI Adapter (14100401)
Hardware Address: 00:04:ac:7c:e2:fb

Transmit Statistics:	Receive Statistics:
----------------------	---------------------

Using the Virtual I/O Server

```
-----  
Packets: 46  
Bytes: 2944  
Interrupts: 43  
Transmit Errors: 0  
Packets Dropped: 0
```

```
-----  
Packets: 46  
Bytes: 2944  
Interrupts: 47  
Receive Errors: 0  
Packets Dropped: 0  
Bad Packets: 0
```

```
Max Packets on S/W Transmit Queue: 3  
S/W Transmit Queue Overflow: 0  
Current S/W+H/W Transmit Queue Length: 1
```

```
Broadcast Packets: 44  
Multicast Packets: 2  
No Carrier Sense: 0  
DMA Underrun: 0  
Lost CTS Errors: 0  
Max Collision Errors: 0  
Late Collision Errors: 0  
Deferred: 0  
SQE Test: 0  
Timeout Errors: 0  
Single Collision Count: 0  
Multiple Collision Count: 0  
Current HW Transmit Queue Length: 1
```

```
Broadcast Packets: 44  
Multicast Packets: 2  
CRC Errors: 0  
DMA Overrun: 0  
Alignment Errors: 0  
No Resource Errors: 0  
Receive Collision Errors: 0  
Packet Too Short Errors: 0  
Packet Too Long Errors: 0  
Packets Discarded by Adapter: 0  
Receiver Start Count: 0
```

General Statistics:

```
-----  
No mbuf Errors: 0  
Adapter Reset Count: 0  
Adapter Data Rate: 2000  
Driver Flags: Up Broadcast Running  
              Simplex Promiscuous AlternateAddress  
              64BitSupport PrivateSegment DataRateSet
```

Adapter Specific Statistics:

```
-----  
Additional Driver Flags: Autonegotiate  
Entries to transmit timeout routine: 0  
Firmware Level: 13.0.10  
Transmit and Receive Flow Control Status: Enabled  
Link Status: Up  
Autonegotiation: Enabled  
Media Speed Running: 1000 Mbps Full Duplex
```

Correcting failed, shared, Ethernet adapter configuration

When you configure a shared Ethernet adapter the task can fail with the following error:

```
Method error (/usr/lib/methods/cfgsea):  
0514-040 Error initializing a device into the kernel.
```

Do the following:

- Verify that the physical and virtual adapters that are being used to create the shared Ethernet device are available by running the following command:

```
lsdev -type adapter
```

- Make sure that the physical adapter is not configured. Run the following command:

```
netstat -state
```

The adapter must *not* show in the output.

- Verify that the virtual adapters used are trunk adapters by running the following command:

```
entstat -all entX | grep Trunk
```

Debugging problems with Connectivity

To help debug problems with connectivity, follow these steps:

1. Verify that the source client partition can ping another client partition on the same system without going through the Virtual I/O Server. If this fails, the problem is likely in the client partition's virtual Ethernet setup. Otherwise, proceed to the step 2.

2. Start a ping on the source partition to a destination machine so that the packets are sent through the Virtual I/O Server. This ping will most likely fail. Proceed to the next step with the ping test running.
3. On the Virtual I/O Server, do the following:
 - a. Type the following command, where *sea_adapter* is your shared Ethernet adapter:


```
entstat all sea_adapter
```
 - b. Verify the VLAN ID that the partition belongs to is associated with the correct virtual adapter in the VLAN IDs section of the output. Examine the ETHERNET STATISTICS for the virtual adapter for this VLAN and verify that the packet counts under the Receive statistics are increasing. This verifies that the packets are being received by the Virtual I/O Server through the correct adapter. If this is not happening, the problem might be in the virtual adapter configuration. Verify the VLAN ID information for the adapters by using the HMC.
 - c. Examine the ETHERNET STATISTICS for the physical adapter for this VLAN and verify that the packet counts under the Transmit statistics are increasing. This step verifies that the packets are being sent out of the Virtual I/O Server. If this count is not increasing, the packets are not going out of the physical adapter, and further debugging requires the system trace utility. Follow the instructions in Step 3.g to collect a system trace, statistical information, and the configuration description. Contact service and support if you need to debug further. See [Customer service, support, and troubleshooting](#) for information about service and support.
 - d. Verify that the target system outside (on physical side of Virtual I/O Server) is receiving packets and sending out replies. If this is not happening, either the wrong physical adapter is associated with the shared Ethernet adapter or the Ethernet switch might not be configured correctly.
 - e. Examine the ETHERNET STATISTICS for the physical adapter for this VLAN and verify that the packet counts under the Receive statistics column are increasing. This step verifies that the ping replies are being received by the Virtual I/O Server. If this count is not increasing, the switch might not be configured correctly.
 - f. Examine the ETHERNET STATISTICS for the virtual adapter for this VLAN and verify that the packet counts under the Transmit statistics are increasing. This step verifies that the packet is being transmitted by the Virtual I/O Server through the correct virtual adapter. If this count is not increasing, start the system trace utility. Follow the instructions in the following step to collect a system trace, statistical information, and the configuration description. Work with customer support to debug this further.
 - g. Use the Virtual I/O Server trace utility to debug connectivity problems. A system trace can be started using the startrace command specifying the trace hook ID. The trace hook ID for shared Ethernet adapter is 48F. The stoptrace command is used to stop the trace. The cattracerpt command reads the trace log, formats the trace entries, and writes a report to standard output.

Parent topic: [Troubleshooting the Virtual I/O Server logical partition](#)

Troubleshooting the client logical partition

Find information and procedures for troubleshooting the client partitions.

If your client partition is using virtual I/O resources, check the Service Focal Point and Virtual I/O Server first to ensure that the problem is not on the server. See sections on [Troubleshooting](#), [Service Focal Point](#), and [Troubleshooting Virtual I/O Server problems](#).

Troubleshooting Virtual SCSI problems on the client logical partition

The following information can help identify Virtual SCSI problems on the client logical partition.

On client partitions running the current level of AIX, when a hardware error is logged on the server and a corresponding error is logged on the client partition, the Virtual I/O Server provides a correlation error message in the error report.

Run the following command to gather an error report:

```
errpt -a
```

Running the errpt command returns results similar to the following:

```

LABEL:          VSCSI_ERR2
IDENTIFIER:     857033C6

Date/Time:      Tue Feb 15 09:18:11 2005
Sequence Number: 50
Machine Id:     00C25EEE4C00
Node Id:        vio_client53A
Class:          S
Type:           TEMP
Resource Name:  vscsi2

```

```

Description
Underlying transport error

```

```

Probable Causes
PROCESSOR

```

```

Failure Causes
PROCESSOR

```

```

Recommended Actions
PERFORM PROBLEM DETERMINATION PROCEDURES

```

```

Detail Data
Error Log Type
01
Reserve
00
Error Number
0006
RC
0000 0002
VSCSI Pointer

```

Compare the LABEL, IDENTIFIER, and Error Number values from your error report to the values in the following table to help identify the problem and determine a resolution:

Table 1. Labels, identifiers, error numbers, problem descriptions, and resolutions of common Virtual SCSI client partition problems

Label	Identifier	Error Number	Problem	Resolution
VSCSI_ERR2	857033C6	0006 RC 0000 0002	The Virtual SCSI server adapter on the Virtual I/O Server partition is not open.	Make the server adapter on the Virtual I/O Server partition available for use.
		001C RC 0000 0000	The Virtual SCSI server adapter on the Virtual I/O Server partition has been closed abruptly.	Determine why the server adapter in the Virtual I/O Server partition was closed.
VSCSI_ERR3	ED995F18	000D RC FFFF FFF0	The Virtual SCSI server adapter on the Virtual I/O Server partition is being used by another client.	Terminate the client partition that is using the server adapter.
		000D RC FFFF FFF9	The Virtual SCSI server adapter (partition number and slot number) specified in the client adapter definition does not exist.	On the HMC, correct the client adapter definition to associate it with a valid server adapter.

- **Recovering from disks not appearing in SMS**

Learn how to recover from disks not showing in the System Management Services (SMS) menu when trying to boot or install a client logical-partition.

Parent topic: [Troubleshooting the Virtual I/O Server](#)

Recovering from disks not appearing in SMS

Learn how to recover from disks not showing in the System Management Services (SMS) menu when trying to boot or install a client logical-partition.

Occasionally, the disk that is needed to install the client logical-partition cannot be located. In this situation, if the client is already installed, start the client logical-partition. Ensure that you have the latest levels of the software and firmware. Then ensure that the Slot number of the virtual SCSI server adapter matches the **Remote partition virtual slot number** of the virtual SCSI client adapter.

1. Ensure that you have the latest levels of the Hardware Management Console, firmware, and Virtual I/O Server. Follow these steps:
 - a. To check whether you have the latest level of the HMC, see [Getting HMC machine code fixes](#).
 - b. To check whether you have the latest firmware, see [Getting server firmware and power subsystem firmware fixes](#).
2. Ensure the server virtual SCSI adapter slot number is mapped correctly to the client logical-partition remote slot number. Follow these steps:
 - a. On the HMC, right-click the server profile.
 - b. Click Properties.
 - c. Click the Virtual I/O Server tab.
 - d. If the Only selected remote partition and slot can connect radio button is not selected, select it.
 - e. Note the **Remote partition** and **Remote partition virtual slot number** values. This shows the client logical-partition name and the client logical-partition virtual slot number. This is the client logical-partition and slot number that can connect to the slot given in the Slot number dialogue box at the top of the Virtual SCSI Adapter Properties window.
 - f. Repeat items a through e in this step for the client logical-partition.
3. The Slot number value on the client logical-partition must match the Remote partition virtual slot number on the Virtual I/O Server partition, and the Slot number value on the Virtual I/O Server partition must match the Remote partition virtual slot number on the client logical-partition. If these numbers do not match, from the HMC, modify the profile properties to reflect the correct mapping.
4. From the Virtual I/O Server command line, type `cfgdev`.
5. Shut down and reactivate the client logical-partition.
6. From the Virtual I/O Server command line, type `lsmap -all`. You see results similar to the following:

SVSA	Physloc	Client Partition ID
vhost0	U9113.550.10BE8DD-V1-C3	0x00000002
VTD	vhdisk0	
LUN	0x8100000000000000	
Backing device	hdisk5	
Physloc	U787B.001.DNW025F-P1-C5-T1-W5005076300C10899-L536F00000000000	

In this example, the client partition ID is 2 (0x00000002).

Note: If the client partition is not yet installed, the Client Partition ID is 0x00000000.

The slot number of the server SCSI adapter is displayed under Physloc column. The digits following the `-C` specify the slot number. In this case, the slot number is 3.

7. From the Virtual I/O Server command line, type `lsdev -virtual`. You see results similar to the following:

name	status	description
vhost0	Available	Virtual SCSI Server Adapter
vhdisk0	Available	Virtual Target Device - Disk

The virtual SCSI server adapter and the virtual target device are shown in the Available state column.

Parent topic: [Troubleshooting the client logical partition](#)

Related information for the Virtual I/O Server

Find other information related to the Virtual I/O Server.

The following Web sites and ESCALA Power5 Hardware Information topics relate to the Virtual I/O Server topic.

Other information

- [Managing your server](#)
- [Customer service and support](#)

Saving PDF files

To save a PDF on your workstation for viewing or printing, use the following task:

1. Right-click the link to the PDF file in your browser.
2. Click Save Target As... if you are using Internet Explorer. Click Save Link As... if you are using Netscape Communicator.
3. Navigate to the directory in which you would like to save the PDF.
4. Click Save.

Downloading Adobe Reader

You need Adobe Reader to view or print these PDFs. You can download a copy from the [Adobe Web site](http://www.adobe.com/products/acrobat/readstep.html) (www.adobe.com/products/acrobat/readstep.html) .

Parent topic: [Using the Virtual I/O Server](#)

Virtual I/O Server command descriptions

This topic includes descriptions of the Virtual I/O Server commands.

A description of each Virtual I/O Server command is given in this topic. This information is also available from the Virtual I/O Server command line using the **man** command. A printable version of these commands is available in the [Virtual I/O Server commands reference](#).

This topic is divided into the following sections:

- [Installation commands](#)
- [Volume group commands](#)
- [Logical volume commands](#)
- [Physical volume commands](#)
- [Storage pool commands](#)
- [Network commands](#)
- [Device commands](#)
- [User ID commands](#)
- [Security commands](#)
- [Maintenance commands](#)
- [Workload manager commands](#)
- [Integrated Virtualization Manager commands](#)
- [Standard shell commands](#)

Installation commands

- [installios Command](#)
- [ioslevel Command](#)
- [license Command](#)
- [lssw Command](#)
- [oem_platform_level Command](#)
- [oem_setup_env Command](#)
- [remote_management Command](#)
- [updateios Command](#)

Volume group commands

- [activatevg Command](#)
- [chvg Command](#)
- [deactivatevg Command](#)
- [exportvg Command](#)
- [extendvg Command](#)
- [importvg Command](#)
- [lsvg Command](#)
- [mirrorios Command](#)
- [mkgv Command](#)
- [redefvg Command](#)
- [reducevg Command](#)
- [syncvg Command](#)
- [unmirrorios Command](#)

Logical volume commands

- [chlv Command](#)
- [cplv Command](#)
- [extendlv Command](#)
- [lslv Command](#)
- [mklv Command](#)
- [mklvcopy Command](#)
- [rmlv Command](#)
- [rmlvcopy Command](#)

Physical volume commands

- [lspv Command](#)
- [migratepv Command](#)

Storage pool commands

- [chsp Command](#)
- [lssp Command](#)

- mksp Command
- mkbdsp Command
- rmbdsp Command

Network commands

- cfmnagg Command
- cfgnamesrv Command
- chtcpip Command
- entstat Command
- hostmap Command
- hostname Command
- lsnetsh Command
- lstcpip Command
- mktcpip Command
- netstat Command
- optimizenet Command
- ping Command
- rmtcpip Command
- startnetsh Command
- stopnetsh Command
- traceroute Command

Device commands

- cfgdev Command
- chdev Command
- chpath Command
- lsdev Command
- lsmap Command
- lspath Command
- mkpath Command
- mkvdev Command
- rmdev Command
- rmpath Command
- rmvdev Command

User ID commands

- chuser Command
- lsuser Command
- mkuser Command
- passwd Command
- rmuser Command

Security commands

- [lsfailedlogin Command](#)
- [lsqcl Command](#)
- [viosecure Command](#)

Workload manager commands

- [wkldagent Command](#)
- [wkldmgr Command](#)
- [wkldout Command](#)

Maintenance commands

- [backupios Command](#)
- [bootlist Command](#)
- [cattracerpt Command](#)
- [chdate Command](#)
- [chlang Command](#)
- [diagmenu Command](#)
- [errlog Command](#)
- [fscck Command](#)
- [invscout Command](#)
- [ldfware Command](#)
- [loginmsg Command](#)
- [lsfware Command](#)
- [lsparinfo Command](#)
- [motd Command](#)
- [mount Command](#)
- [pdump Command](#)
- [restorevgstruct Command](#)
- [savevgstruct Command](#)
- [showmount Command](#)
- [shutdown Command](#)
- [snap Command](#)
- [startsysdump Command](#)
- [starttrace Command](#)
- [stoptrace Command](#)
- [sysstat Command](#)
- [topas Command](#)
- [unmount Command](#)
- [viostat Command](#)

Integrated Virtualization Manager commands

- [bkprofdata Command](#)
- [chled Command](#)
- [chlparutil Command](#)
- [chsvcevent Command](#)
- [chsyscfg Command](#)
- [chsysstate Command](#)
- [lpcfgop Command](#)

- lshwres Command
- lsled Command
- lsparutil Command
- lsrefcode Command
- ,lssvcevents Command
- lssyscfg Command
- lssysconn Command
- mkgencfg Command
- mksvcevent Command
- mksyscfg Command
- mkvt Command
- rmsyscfg Command
- rmvt Command
- rstprofdata Command

Standard shell commands

- awk Command
- cat Command
- chmod Command
- clear Command
- cp Command
- crontab Command
- date Command
- ftp Command
- grep Command
- head Command
- ls Command
- man Command
- mkdir Command
- more Command
- mv Command
- ping Command
- rm Command
- sed Command
- stty Command
- tail Command
- tee Command
- vi Command
- wall Command
- wc Command
- who Command

Parent topic: [Using the Virtual I/O Server](#)

Technical publication remarks form

Title :	ESCALA POWER5 Hardware Information Using the Virtual I/O Server
----------------	---

Reference N° :	86 A1 24EW 00
-----------------------	---------------

Date:	July 2006
--------------	-----------

ERRORS IN PUBLICATION

--

SUGGESTIONS FOR IMPROVEMENT TO PUBLICATION

--

Your comments will be promptly investigated by qualified technical personnel and action will be taken as required.
If you require a written reply, please include your complete mailing address below.

NAME : _____ Date : _____

COMPANY : _____

ADDRESS : _____

Please give this technical publication remarks form to your BULL representative or mail to:

Bull - Documentation Dept.
1 Rue de Provence
BP 208
38432 ECHIROLLES CEDEX
FRANCE
info@frec.bull.fr

Technical publications ordering form

To order additional publications, please fill in a copy of this form and send it via mail to:

BULL CEDOC
357 AVENUE PATTON
B.P.20845
49008 ANGERS CEDEX 01
FRANCE

Phone: +33 (0) 2 41 73 72 66
FAX: +33 (0) 2 41 73 70 66
E-Mail: srv.Duplicopy@bull.net

CEDOC Reference #	Designation	Qty
-- -- []		
-- -- []		
-- -- []		
-- -- []		
-- -- []		
-- -- []		
-- -- []		
-- -- []		
-- -- []		
-- -- []		
-- -- []		
-- -- []		
-- -- []		
-- -- []		
[] : The latest revision will be provided if no revision number is given.		

NAME: _____ Date: _____

COMPANY: _____

ADDRESS: _____

PHONE: _____ FAX: _____

E-MAIL: _____

For Bull Subsidiaries:

Identification: _____

For Bull Affiliated Customers:

Customer Code: _____

For Bull Internal Customers:

Budgetary Section: _____

For Others: Please ask your Bull representative.

BULL CEDOC
357 AVENUE PATTON
B.P.20845
49008 ANGERS CEDEX 01
FRANCE

REFERENCE
86 A1 24EW 00