Solaris™ Fibre Channel and Storage Multipathing Administration Guide

Sun Microsystems, Inc.
www.sun.com

Please
Recycle

Adobe PostScript™

Contents

Tables

Preface

The *Solaris Fibre Channel and Storage Multipathing Administration Guide* provides an overview of the Fibre Channel and Storage Multipathing software for the Solaris^{TM} 10 Operating System This guide includes instructions about how to install and configure the software and devices.

This guide is intended for system, storage, and network administrators who create and maintain storage area networks (SANs) and have a high level of expertise in the management and maintenance of SANs.

Note – The Traffic Manager Multipathing functionality has been incorporated into the Solaris Operating System. This document refers only to the multipathing functionality and not to the Traffic Manager product.

Before You Read This Book

Before you read this book, read the Solaris 10 Operating System 3/05 Release Notes.

How This Book Is Organized

Chapter 1 provides an overview of *S*olaris Fibre Channel and Storage Multipathing software.

Chapter 2 provides an overview of configuring the Solaris Fibre Channel and Storage Multipathing software.

Chapter 3 explains how to configure multipathing feature for FC devices provided in the Solaris Operating System.

Chapter 4 provide an overview of steps used to configure SAN devices.

Chapter 5 explains the considerations to configure IPFC SAN Devices.

Appendix A explains Manual configuration for Fabric Connected Devices.

Appendix B provides details on Supported FC-HBA API.

Appendix C provides trouble shooting information for potential problems that may occur while you run the Solaris Fibre Channel and Storage Multipathing software.

Using UNIX Commands

This document might not contain information on basic UNIX® commands and procedures such as shutting down the system, booting the system, and configuring devices.

See one or more of the following for this information:

- *Solaris Handbook for Sun Peripherals*
- AnswerBook2™ online documentation for the Solaris™ Operating System

Shell Prompts

Shell	Prompt
C shell	*machine-name*%
C shell superuser	*machine-name*#
Bourne shell and Korn shell	$
Bourne shell and Korn shell superuser	#

Typographic Conventions

Typeface*	Meaning	Examples
`AaBbCc123`	The names of commands, files, and directories; on-screen computer output	Edit your `.login` file. Use `ls -a` to list all files. `% You have mail.`
`AaBbCc123`	What you type, when contrasted with on-screen computer output	`% `**`su`** `Password:`
AaBbCc123	Book titles, new words or terms, words to be emphasized. Replace command-line variables with real names or values.	Read Chapter 6 in the *User's Guide*. These are called *class* options. You *must* be superuser to do this. To delete a file, type `rm` *filename*.

* The settings on your browser might differ from these settings.

Related Documentation

Application	Title	Part Number
Solaris Fibre Channel and Storage Multipathing Administration Guide	*Solaris 10 Operating System 3/05 Release Notes*	817-7741-10

Accessing Sun Documentation

You can view, print, or purchase a broad selection of Sun documentation, including localized versions, at: `http://www.sun.com/documentation`

Third-Party Web Sites

Sun is not responsible for the availability of third-party web sites mentioned in this document. Sun does not endorse and is not responsible or liable for any content, advertising, products, or other materials that are available on or through such sites or resources. Sun will not be responsible or liable for any actual or alleged damage or loss caused by or in connection with the use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

Contacting Sun Technical Support

If you have technical questions about this product that are not answered in this document, go to:

`http://www.sun.com/service/contacting`

Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions. You can submit your comments by going to:

`http://www.sun.com/hwdocs/feedback`

Please include the title and part number of your document with your feedback:

Solaris Fibre Channel and Storage Multipathing Administration Guide, part number 817-7741-10

Product Overview

This chapter provides an overview of the Solaris Fibre Channel (FC) and Storage Multipathing software. This information will prove helpful during installation and configuration of the software.

The Solaris FC and Storage Multipathing software is integrated in the Solaris 10 OS. Whether the system that is being upgraded has prior version of the Solaris FC and Storage Multipathing software installed or not, there is no need to separately install the software package during or after the Solaris 10 OS installation. That applies to the Solaris 10 OS new (initial) install as well.

In Solaris 10 OS, fabric-connected devices are configured and made available to the host automatically during install and boot time.

Note – If performing an upgrade and if you want to make any FC devices unavailable after upgrade, those devices have to be manually unconfigured using "`cfgadm -c unconfigure`" commands. However, to make those devices permanently unavailable to the system, you may want to consider using switch zoning and/or LUN masking. The changes made by `cfgadm` unconfigure will not persist after a reboot unless manual configuration for fabric-connected devices has been turned on. To disable the fabric devices discovery during boot or install, please refer to Appendix A titled, "Manual Configuration for Fabric Connected Devices" on page 37.

This chapter contains the following sections:

- "Software Features" on page 2
- "Supported Standards" on page 4

Software Features

In general, the Solaris Fibre Channel and Storage Multipathing software enables FC connectivity for the Solaris hosts. The software resides on the server and identifies the storage and switch devices on your SAN. It allows you to attach either loop or fabric SAN storage devices while providing a standard interface with which to manage them.

Fibre Channel Software Features

The Solaris FC and Storage Multipathing software provides the following key features:

- **Dynamic Storage Discovery** – The Solaris FC and Storage Multipathing software automatically recognizes devices and any modifications made to device configurations. This makes devices available to the system without requiring you to reboot or manually change information in configuration files.
- **Persistent Device Naming** – Devices that are configured within the Solaris FC and Storage Multipathing software maintain their device naming through reboots and/or reconfiguration. The only exception to this are tape devices, found in `/dev/rmt`, will not change unless they are removed and later regenerated.
- **FCAL Support** – OpenBoot^{TM} Prom (OBP) commands that are used on servers can access Fibre Channel-Arbitrated Loop (FCAL) attached storage for scanning the FC loop.
- **Fabric Booting** – Sun supported 1 Gbit and 2 Gbit host bus adapters (HBAs) can boot from fabric devices as well as non-fabric devices. Fabric topologies with FC switches provide higher speed, more connections, and port isolation. Fabric booting is available only on SPARC^R.
- **FC-HBA Library** – What was previously known as the Storage Networking Industry Association (SNIA) Fibre Channel HBA(FC-HBA) library is now known as the FC-HBA library. The FC-HBA library application programming interface (API) enables management of FC HBAs and provides a standards-based interface for other applications (such as Sun StorEdge^{TM} Enterprise Storage Manager) that can be used to gather information about FC HBAs.

Man pages for common FC-HBA APIs are included in the Solaris 10 Operating System. For additional information on FC specifications refer to `http://www.t11.org`.

Multipathing Software Features With Fibre Channel Storage

The Solaris 10 OS contains the Solaris FC and Storage Multipathing software. This is enabled by default for x86-based platforms and optional for SPARC-based platforms. The software contains the following features:

- **Path Management** – The Solaris FC and Storage Multipathing software dynamically manages the paths to any storage devices it supports. The addition or removal of paths to a device is done automatically when a path is brought online or removed from a service. This allows hosts configured with the Solaris FC and Storage Multipathing software to begin with a single path to a device and add more host controllers, increasing bandwidth and RAS, without changing device names or modifying applications. For Sun storage, there are no configuration files to manage or databases to keep current. For Storage other than Sun, contact your storage vendor for methods to enable support and to ensure it is qualified with the multipathing software.
- **Single Device Instances** - Unlike other multipathing solutions, the Solaris FC and Storage Multipathing software is fully integrated with the Solaris 10 OS. This allows the Solaris FC and Storage Multipathing software to display multipath devices as single device instances instead of as one device, or device link, per path. This reduces the cost of managing complex storage architectures, since it enables utilities, such as `format`(1M) or higher level applications such as the Solaris Volume Manager, to "see" one representation of a storage device instead of a separate device for each path.
- **Failover Support** – Implementing higher levels of RAS requires redundant host connectivity to storage devices. The Solaris FC and Storage Multipathing software manages the failure of storage paths while maintaining host I/O connectivity through available secondary paths.
- **Symmetrical/Asymmetrical Device Support** – The Solaris FC and Storage Multipathing software support the following disk storage devices.
  - All Sun disk storage - both symmetric and asymmetric
  - All T10/T11 standards complaint non-Sun symmetrical disk devices
  - Many non-Sun Asymmetrical disk arrays.

Please contact your storage vendor to see if your device is supported.

In addition, T10 ALUA support has been added for those asymmetrical devices that support this T10 standard.

- **I/O Load Balancing** – In addition to providing simple fail over support, the Solaris FC and Storage Multipathing software can use any active paths to a storage device to send and receive I/O. With I/O routed through multiple host connections, bandwidth can be increased by the addition of host controllers. The Solaris FC and Storage Multipathing software uses a round-robin load-balancing algorithm, by which individual I/O requests are routed to active host controllers in a series, one after the other.

- **Queue Depth** – SCSI storage arrays present storage to a host in the form of a LUN. LUNs have a finite set of resources available, such as the amount of data that can be stored, as well as the number of active commands that a device or LUN can process at one time. The number of active commands that can be issued before a device blocks further I/O is know as queue depth. When the Solaris FC and Storage Multipathing software is enabled, a single queue is created for each LUN regardless of the number of distinct or separate paths it may have to the host.This allows the disk driver to maintain and balance 1 queue to the LUN, effectively managing queue depth. No other multipathing softwares available for the Solaris 10 OS can claim this.
- `stmsboot` **Command** – The Solaris 10 OS cannot be installed initially onto the Solaris multipathing software devices. However, the `stmsboot` command, which is included with the Solaris FC and Storage Multipathing software, allows you to enable the Solaris Multipathing software on the boot device after the installation has been completed for SPARC devices.
- **Dynamic Reconfiguration** – The Solaris FC and Storage Multipathing Software supports the Solaris 10 OS Dynamic Reconfiguration (DR).

Supported Standards

The Solaris FC and Storage Multipathing software is based on open standards for communicating with devices and device management, ensuring interoperability with other standards-based devices and software. The following standards are supported by the Solaris FC and Storage Multipathing software:

- T10 standards, including SCSI-3, SAM, FCP, SPC, and SBC
- T11.3 FC standards, including FC-PH, FC-AL, FC-LS, and FC-GS
- T11.5 storage management standards, including FC-HBA
- IETF standards, including RFC 2625

Configuration Overview

This chapter provides information about configuring the Solaris FC and Storage Multipathing software.

This chapter contains the following sections:

- "Configuration Task Overview" on page 5
- "Configuration Considerations" on page 6

Configuration Task Overview

This section provides a high level overview of the tasks that you should perform to configure the Solaris FC and Storage Multipathing software.

The following table lists the configuration tasks, description about each task and the reference information where you can find the procedures to perform each task.

TABLE 2-1   Tasks to Configure Solaris FC and Storage Multipathing Software

Configuration Task	Task Description	Reference Information
Enable Multipathing	Multipathing capability is provided by the Solaris FC and Storage Multipathing software in the Solaris 10 OS. This software enables multipathing: * By default on x86 platforms. * By manual configuration on SPARC platforms.	The steps for enabling multipathing are outlined in "Configuring Multipathing" on page 9.
Configure Devices	In the Solaris 10 OS, FCAL and fabric-connected devices are made available to the host. This differs from previous versions of the multipathing software running on the Solaris 8 or 9 OS. In those versions, manual configuration steps were required to achieve fabric-connected device availability on the Solaris host.	Please refer to the chapter "Configuring SAN Devices" on page 25 for more information.
Set up FC Boot Device	The Solaris FC and Storage Multipathing software allows a Solaris SPARC server to boot from a fabric device. Note: This is not available on x86 platforms.	Please refer to the section "Configuring Fabric Boot Devices on SPARC" on page 27 for more information.
Configure IPFC SAN devices	You can configure host recognition of IPFC devices and implementation of IP over FC in a SAN. Configuration of IPFC depends on the instance of the Fabric Port (FP) driver for the FC adapter ports.	This information is outlined in "Configuring IPFC SAN Devices" on page 29.

Configuration Considerations

Before you configure the Solaris FC and Storage Multipathing software, consider the following:

■ Configure ports and zones according to the vendor-specific documentation for storage and switches.

- LUN masking enables specific LUNs to be seen by specific hosts. See your vendor-specific storage documentation that describes masking.
- Power Management needs to be disabled for hosts and devices on a SAN. See `power.conf` (4) for details about power management.

Configuring Multipathing Software

This chapter explains how to configure multipathing feature for FC devices provided in the Solaris 10 OS. It also provides the considerations while enabling or disabling multipathing feature on SPARC based systems, x86 based systems, per-port basis and third-party storage devices.

This chapter contains the following sections:

Configuring Multipathing

The multipathing feature for FC devices can be configured to control all Sun supported FC HBAs. Multipathing is disabled by default for FC devices on SPARC based systems, but is enabled by default on x86 based systems. Configuration of the multipathing software depends on how you intend to use your system.

Note – The multipathing feature is not available for parallel SCSI devices but is available for FC disk devices. Multipathing is not supported on tape drives or libraries, or on IP over FC.

Enabling or Disabling Multipathing

Before you change multipathing configuration consider the following and then follow the instructions for your machine architecture (SPARC or x86) described in the subsequent sections.

- **Device Specific Considerations**

Some devices need to be properly configured to work with the multipathing software. Refer to your storage array documentation for details on the device specific configuration for your device.

- **Device Name Change Considerations**

In the `/dev` and `/devices` trees, multipath-enabled devices receive new names that indicate that they are under multipath control. This means a device will have a different name from its original name when it is under multipath control.

Example device name with multipath disabled:

```
/dev/dsk/c1t1d0s0
```

Example device name with multipath enabled:

```
/dev/dsk/c3t2000002037CD9F72d0s0
```

Because of this, if you have applications that directly consume device names, they must be configured to use the new names whenever you change multipath configuration from disabled to enabled or vice-versa.

The system's `/etc/vfstab` file and the dump configuration also contains references to device names. On SPARC based systems, the `stmsboot`(1M) command described in the following sections automatically updates the `/etc/vfstab` and the dump configuration with the new device names. On x86 based systems this must be performed manually. You need to do this only if the device names listed in `/etc/vfstab` or your dump device are affected by the device name changes. For updating `/etc/vfstab`, simply edit the file with any text editor and substitute new device names in place of old device names. Use `dumpadm`(1M) command to update the dump configuration.

Enabling or Disabling Multipathing on SPARC Based Systems

On SPARC based systems, you can enable or disable multipathing on FC devices using the `stmsboot`(1M) command. The command also updates `/etc/vfstab` and dump configuration to reflect devices name changes during the next reboot.

The following consideration apply (for `stmsboot -e`, `-d`, and `-u` options):

- You should plan to reboot immediately after running the `stmsboot` command.
- Ensure that `eeprom`(1M) boot-device is set to boot from your current boot device. This is needed as `stmsboot` command reboots the machine to complete the operation.
- The `stmsboot` command saves a copy of your original `/kernel/drv/fp.conf` and `/etc/vfstab` files before modifying them. The saved files can be used to recover from any unexpected problems. See the Appendix - "Multipathing Troubleshooting" on page 61, if you encounter unexpected problems during `stmsboot`.

Note – The behavior of `stmsboot` command in the Solaris 10 differs from the `stmsboot` command provided by Sun StorEdge SAN Foundation software in the Solaris 8 and 9 OS. In the Solaris 8 and 9 OS, `stmsboot` command is used to enable or disable multipathing on the boot device only. In the Solaris 10, the command is used to enable or disable multipathing on all devices. The `stmsboot`(1M) is available only on SPARC based systems.

▼ To Enable Multipathing on SPARC Based Systems

Perform the following steps to enable multipathing on all FC devices on SPARC based systems. If you want to enable multipathing only on specific FC ports refer to the section "Enabling or Disabling Multipathing on a Per-Port Basis" on page 14.

1. **The multipathing software automatically recognizes Sun supported devices. If you want to enable multipathing on third-party devices, add entries to** `/kernel/drv/scsi_vhci.conf` **file as mentioned in the section "Device Name Change Considerations" on page 10.**

2. **Run the** `stmsboot -e` **command:**

```
# stmsboot -e
WARNING: This operation will require a reboot.
Do you want to continue ? [y/n] (default: y) y
The changes will come into effect after rebooting the system.
Reboot the system now ? [y/n] (default: y) y
```

3. **You are prompted to reboot. During the reboot,** `/etc/vfstab` **and the dump configuration will be updated to reflect the device name changes.**

   After the reboot, if necessary, configure your applications to use new device names as described in "Device Name Change Considerations" on page 10.

▼ To Disable Multipathing on SPARC Based Systems

Perform the following steps to disable multipathing on all FC devices on SPARC based systems, if you have enabled it before. If you want to disable multipathing only on specific FC ports refer to the section "Enabling or Disabling Multipathing on a Per-Port Basis" on page 14.

1. **Run the** `stmsboot -d` **command:**

```
# stmsboot -d
WARNING: This operation will require a reboot.
Do you want to continue? [y/n] (default: y) y
The changes will come into effect after rebooting the system.
Reboot the system now? [y/n] (default: y) y
```

2. **You are prompted to reboot. During the reboot,** `/etc/vfstab` **and the dump configuration will be updated to reflect the device name changes.**

   After the reboot, if necessary, configure your applications to use new device names as described in "Device Name Change Considerations" on page 10.

Enabling or Disabling Multipathing on x86 Based Systems

Multipathing is enabled by default on x86 based systems. Use the following steps to disable multipathing or to enable multipathing if you have disabled it previously.

Caution – If the critical file systems needed for booting, such as the `root (the '/')`, `/usr`, or `/var` file systems, are on FC disks, you should not disable multipathing on them otherwise your system will not boot. You may disable multipathing on all FC devices if the critical file systems are not on FC disks. If the critical file systems are on FC disks, you may disable multipathing only on those FC HBA controller ports which aren't connected to the disks containing the critical file systems using the instructions in the section "Enabling or Disabling Multipathing on a Per-Port Basis" on page 14.

---

1. **The multipathing software automatically recognizes Sun supported devices. If you want to enable multipathing on third-party devices, add entries to** `/kernel/drv/scsi_vhci.conf` **file as mentioned in the section T"To Configure Third-Party Devices" on page 21.**

2. **Open the** `/kernel/drv/fp.conf` **file in a text editor**

3. **If you want to enable multipathing:**

   Change the value of `mpxio-disable` to "no". If the entry doesn't exist add a new entry.

   ```
   mpxio-disable="no";
   ```

   If you want to disable multipathing:

   Change the value of `mpxio-disable` to "yes". If the entry doesn't exist add a new entry.

   ```
   mpxio-disable="yes";
   ```

Note – This will not affect your per-port multipathing configuration. For information on per-port configuration refer to the section "Enabling or Disabling Multipathing on a Per-Port Basis" on page 14.

---

4. **Save and exit the** `fp.conf` **file.**

5. **Perform a reconfiguration reboot.**

```
# touch /reconfigure
# shutdown -g0 -y -i6
```

6. **If necessary, perform device name updates as described in "Device Name Change Considerations" on page 10.**

Enabling or Disabling Multipathing on a Per-Port Basis

Multipathing can be enabled or disabled on specific FC Host Bus Adapter (HBA) controller ports for finer control. If you enable multipathing on a specific HBA controller port, all supported devices connected to the controller port will have multipathing enabled on them.

The procedure described below applies for both SPARC and x86 based machines.

Considerations for Per-Port Configuration

Before you start configuring the software by port, consider the following:

■ You use the `/kernel/drv/fp.conf` file to set both global and per-port multipath settings. Per-port multipath settings have priority over the global setting. That means, if global multipathing is enabled but a specific port has been disabled for multipathing, the port will be unable to run multipathing. Conversely, even if global multipathing has been disabled, but a specific port has been enabled for multipathing, the specific port will be enabled for multipathing.

■ Load balancing is controlled by the global load-balance property in `/kernel/drv/scsi_vhci.conf` file and can not be applied on a per-port basis.

■ If a device has more than one path to the host, all paths to the device must be configured with multipathing enabled or disabled.

■ Configuring multipathing by port enables the multipathing software to co-exist with other multipathing solutions like VERITAS Dynamic Multipathing (DMP), or EMC PowerPath. However, devices and paths should not be shared between the multipathing software and other multipathing solutions.

▼ To Configure Multipathing by Port

1. Log in as superuser.

Determine the HBA controller ports that you want the multipathing software to control. For example, to select the desired device, perform an `ls -l` command on `/dev/fc`. The following is an example of the `ls -l` command output.

```
lrwxrwxrwx   1 root    root            49 Apr 17 18:14 fp0 ->
../../devices/pci@6,2000/SUNW,qlc@2/fp@0,0:devctl
lrwxrwxrwx   1 root    root            49 Apr 17 18:14 fp1 ->
../../devices/pci@7,2000/SUNW,qlc@2/fp@0,0:devctl
lrwxrwxrwx   1 root    root            49 Apr 17 18:14 fp2 ->
../../devices/pci@a,2000/SUNW,qlc@2/fp@0,0:devctl
lrwxrwxrwx   1 root    root            49 Apr 17 18:14 fp3 ->
../../devices/pci@b,2000/SUNW,qlc@2/fp@0,0:devctl
lrwxrwxrwx   1 root    root            50 Apr 17 18:14 fp4 ->
../../devices/pci@12,2000/SUNW,qlc@2/fp@0,0:devctl
lrwxrwxrwx   1 root    root            56 Apr 17 18:14 fp5 ->
../../devices/pci@13,2000/pci@2/SUNW,qlc@4/fp@0,0:devctl
lrwxrwxrwx   1 root    root            56 Apr 17 18:14 fp6 ->
../../devices/pci@13,2000/pci@2/SUNW,qlc@5/fp@0,0:devctl
lrwxrwxrwx   1 root    root            56 Apr 17 18:14 fp7 ->
../../devices/sbus@7,0/SUNW,qlc@0,30400/fp@0,0:devctl
```

Note – The fp7 is a SBus HBA. The fp5 and fp6 include two `/pci` elements. This indicates a dual PCI HBA. The rest of the entries do not have additional PCI bridges and are single PCI HBAs.

2. Open the `/kernel/drv/fp.conf` **file and explicitly enable or disable multipathing on an HBA controller port. This file allows you to enable or disable both the global multipath setting, as well as multipath settings for specific ports.**

Change the value of global mpxio-disable property. If the entry doesn't exist add a new entry. The global setting applies to all ports except the ports specified by the per-port entries.

a. To enable multipathing globally, change to

```
mpxio-disable="no";
```

b. To disable multipathing globally, change to

```
mpxio-disable="yes";
```

Add the per-port mpxio-disable entries - one entry for every HBA controller port you want to configure. Per-port settings override the global setting for the specified ports.

a. To enable multipathing on a HBA port, add

```
name="fp" parent="parent name" port=port-number mpxio-disable=
"no";
```

b. To disable multipathing on a HBA port, add

```
name="fp" parent="parent name" port=port-number mpxio-disable=
"yes";
```

The following example disables multipathing on all HBA controller ports except the two specified ports:

```
mpxio-disable="yes";
name="fp" parent="/pci@6,2000/SUNW,qlc@2" port=0 mpxio-disable=
"no";
name="fp" parent="/pci@13,2000/pci@2/SUNW,qlc@5" port=0 mpxio-
disable="no";
```

3. **If running on a SPARC based system, perform the following:**

   Run the `stmsboot -u` command:

   ```
   # stmsboot -u
   WARNING: This operation will require a reboot.
   Do you want to continue ? [y/n] (default: y) y
   The changes will come into effect after rebooting the system.
   Reboot the system now ? [y/n] (default: y) y
   ```

   You are prompted to reboot. During the reboot, `/etc/vfstab` and the dump configuration will be updated to reflect the device name changes.

   If running on a x86 based system, perform a reconfiguration reboot.

   ```
   # touch /reconfigure
   # shutdown -g0 -y -i6
   ```

4. **If necessary, perform device name updates as described in "Device Name Change Considerations" on page 10.**

▼ To Configure an HBA Controller Port on a SPARC System

The following steps show the configuration of a sample single PCI HBA. `fp0 ->../../devices/pci@6,2000/SUNW,qlc@2/fp@0,0:devctl` This sample entry from the `ls -l` command output shown in "To Configure Multipathing by Port" on page 15 indicates a single PCI HBA.

1. **Open the** `/kernel/drv/fp.conf` **file in a text editor**

2. **To explicitly enable multipathing on this HBA port, but disable on the rest of the HBA ports, add the following:**

```
mpxio-disable="yes";
name="fp" parent="/pci@6,2000/SUNW,qlc@2" port=0 mpxio-disable=
"no";
```

To explicitly disable multipathing on this HBA port, but enable on the rest of the HBA ports, add the following:

```
mpxio-disable="no";
name="fp" parent="/pci@6,2000/SUNW,qlc@2" port=0 mpxio-disable=
"yes";
```

3. **Save and exit the file.**

4. **Run the** stmsboot -u **command:**

```
# stmsboot -u
WARNING: This operation will require a reboot.
Do you want to continue ? [y/n] (default: y) y
The changes will come into effect after rebooting the system.
Reboot the system now ? [y/n] (default: y) y
```

You are prompted to reboot. During the reboot, /etc/vfstab and the dump configuration will be updated to reflect the device name changes.

5. **After the reboot, if necessary, configure your applications to use new device names as described in "Device Name Change Considerations" on page 10.**

▼ To Configure an HBA Controller Port on an x86 Based System

The following steps show the configuration of a sample dual PCI HBA. fp6 -> ../../devices/pci@13,2000/pci@2/SUNW,qlc@5/fp@0,0:devctl This sample entry from the ls -l command output shown in "To Configure Multipathing by Port" on page 15 indicates a dual PCI HBA.

1. **Open the** /kernel/drv/fp.conf **file in a text editor**

2. **To explicitly enable multipathing on this HBA port, but disable on the rest of the HBA ports, add the following:**

```
mpxio-disable="yes";
name="fp" parent="/pci@13,2000/pci@2/SUNW,qlc@5" port=0 mpxio-
disable="no";
```

To explicitly disable multipathing on this HBA port, but enable on the rest of the HBA ports, add the following:

```
mpxio-disable="no";
name="fp" parent="/pci@13,2000/pci@2/SUNW,qlc@5" port=0 mpxio-
disable="yes";
```

3. **Save and exit the file.**

4. **Perform a reconfiguration reboot.**

```
# touch /reconfigure
# shutdown -g0 -y -i6
```

5. **If necessary, perform device name updates as described in "Device Name Change Considerations" on page 10.**

Configuring Automatic Failback

Some storage devices have controllers configured as PRIMARY and SECONDARY as part of the array configuration. The SECONDARY paths may operate at a lower performance level than the PRIMARY paths. The multipathing software uses the PRIMARY path to talk to the storage device and keeps the SECONDARY path on standby. In the event of a PRIMARY path failure, the multipathing software automatically directs all I/O traffic over the SECONDARY path, with the PRIMARY path taken offline. This is called a "failover" operation. When the failure associated with the PRIMARY path has been repaired, the multipathing software automatically directs all I/O traffic over the PRIMARY path and keeps the SECONDARY path standby as before. This is called a "failback" operation.

If you want, you may disable the automatic failback operation, in which case, the multipathing software will not automatically failback to the PRIMARY path. Later, after the failure associated with the PRIMARY path has been repaired, you may do a manual failback operation using the luxadm(1M) command.

1. **Open the** `/kernel/drv/scsi_vhci.conf` **file in a text editor.**

2. **To disable automatic failback capability, change the auto-failback entry to:**

    ```
    auto-failback="disable";
    ```

    To enable automatic failback capability (if you disabled it before), change the auto-failback entry to:

    ```
    auto-failback="enable";
    ```

3. **Save and exit the file.**

4. **Reboot the system.**

    ```
    # shutdown -g0 -y -i6
    ```

Configuring Third-Party Storage Devices

Multipathing can be configured on third-party symmetric storage devices. A symmetric storage device is defined as a storage device in which all paths to the storage device are active and I/O commands can be issued through any path.

Note – Before configuring any third-party device, ensure that they are supported. Refer to your third-party user documentation, or third-party vendor for information on proper vendor and product IDs, modes and various settings required for the device to work with multipathing software.

Considerations for Third-Party Device Configuration

Before you configure third-party devices for multipathing, be aware of the following:

■ The device must support the REPORT LUNS SCSI command, and SCSI-3 INQUIRY command VPD Device Identification Page (0x83).

■ You will need the vendor ID and product ID of the device. You can obtain them by using the `format` command followed by the inquiry option on your system. See the `format`(1M) man page.

▼ To Configure Third-Party Devices

Perform the following steps to configure third-party devices if your system already has multipathing enabled. If your system has multipathing disabled, you can configure third-party devices while enabling multipathing as described earlier in this chapter.

1. **Open the** `/kernel/drv/scsi_vhci.conf` **file in a text editor.**

2. **Add the vendor ID and product ID entries.**

The vendor ID and product ID are the vendor and product identification strings that the device returns in SCSI INQUIRY data. The vendor ID must be eight characters long. You must specify all eight characters (even if the trailing characters are spaces).

The product Id can be up to 16 characters long.

```
device-type-scsi-options-list =
"VendorID1ProductID1", "symmetric-option",
"VendorID2ProductID2", "symmetric-option",
...
"VendorIDnProductIDn", "symmetric-option";
symmetric-option = 0x1000000;
```

Note that the entries are separated by the ',' character and the last vendor/product entry is terminated by the ';' character.

For example, to add a device from a "ACME" vendor with a product ID of "MSU", you would add:

```
device-type-scsi-options-list =
"ACME    MSU", "symmetric-option";
symmetric-option = 0x1000000;
```

In addition to "ACME", to add a device from a "XYZ" vendor with a product ID of "ABC", you would add:

```
device-type-scsi-options-list =
"ACME    MSU", "symmetric-option",
"XYZ     ABC", "symmetric-option";
symmetric-option=0x1000000;
```

3. **Save and exit the** `scsi_vhci.conf` **file.**

4. **If running on a SPARC based system, perform the following:**

   Run the `stmsboot -u` command:

   ```
   # stmsboot -u
   WARNING: This operation will require a reboot.
   Do you want to continue ? [y/n] (default: y) y
   The changes will come into effect after rebooting the system.
   Reboot the system now ? [y/n] (default: y) y
   ```

   You are prompted to reboot. During the reboot, `/etc/vfstab` and the dump configuration will be updated to reflect the device name changes.

   If running on a x86 based system, perform a reconfiguration reboot.

   ```
   # touch /reconfigure
   # shutdown -go -y -i6
   ```

5. **If necessary, perform device name updates as described in "Device Name Change Considerations" on page 10.**

Displaying Device Name Changes

The `stmsboot`(1M) command, available only on SPARC based systems, can be used to display the mapping between non-multipath enabled and multipath enabled devices names, after the changes made to the multipath configuration by `stmsboot` `-e`, `-d` or `-u` options take effect. Both the non-multipath enabled and multipath enabled device names must exist in order to show the mapping.

The -L option displays the mapping of devices on all controllers.

```
# stmsboot -L
non-STMS device name                          STMS device name
---------------------------------------------------------------
/dev/rdsk/c2t8d0                  /dev/rdsk/c10t500000E01046DEE0d0
/dev/rdsk/c2t0d0                  /dev/rdsk/c10t500000E01046B070d0
/dev/rdsk/c2t3d0                   /dev/rdsk/c10t20000020372A40AFd0
/dev/rdsk/c2t12d0                  /dev/rdsk/c10t500000E01046DEF0d0
/dev/rdsk/c2t11d0                  /dev/rdsk/c10t500000E01046E390d0
/dev/rdsk/c3t8d0                   /dev/rdsk/c10t500000E01046DEE0d0
/dev/rdsk/c3t0d0                   /dev/rdsk/c10t500000E01046B070d0
/dev/rdsk/c3t3d0                   /dev/rdsk/c10t20000020372A40AFd0
/dev/rdsk/c3t12d0                  /dev/rdsk/c10t500000E01046DEF0d0
/dev/rdsk/c3t11d0                  /dev/rdsk/c10t500000E01046E390d0
```

The -l option displays the mapping of devices on only the specified controller. The following example displays the mapping of controller 3.

```
# stmsboot -l3
non-STMS device name                          STMS device name
---------------------------------------------------------------
/dev/rdsk/c3t8d0                   /dev/rdsk/c10t500000E01046DEE0d0
/dev/rdsk/c3t0d0                   /dev/rdsk/c10t500000E01046B070d0
/dev/rdsk/c3t3d0                   /dev/rdsk/c10t20000020372A40AFd0
/dev/rdsk/c3t12d0                  /dev/rdsk/c10t500000E01046DEF0d0
/dev/rdsk/c3t11d0                  /dev/rdsk/c10t500000E01046E390d0
```

Configuring SAN Devices

This section provides the high level overview of the steps used to configure SAN devices. In Solaris 10, FCAL and fabric-connected devices are made available to the host automatically. This differs from previous versions of the SAN Foundation software running on the Solaris 8 or the Solaris 9 OS. In those versions, manual configuration steps were required to achieve fabric-connected device availability on the host. If you want to manually configure fabric-connected devices, please refer to Appendix A, "Manual Configuration for Fabric Connected Devices" on page 37.

This chapter contains the following sections:

■ "SAN Device Considerations" on page 25
■ "Adding SAN Devices" on page 26
■ "Configuring Fabric Boot Devices on SPARC" on page 27

SAN Device Considerations

You should be aware of the following considerations before configuring the Solaris FC and Storage Multipathing software.

■ Configure ports and zones according to the vendor-specific documentation for storage and switches.
■ With the Solaris 10 you no longer have to manually configure devices unless you enable manual device configuration.
■ LUN masking enables specific LUNs to be seen by specific hosts. See your vendor-specific storage documentation that describes masking.
■ Turn off power management on servers connected to the SAN to prevent unexpected results as one server attempts to power down a device while another attempts to gain access. See `power.conf`(1M) for details about power management.

■ Connect arrays and other storage devices to the SAN with or without multipathing capability. The Solaris FC and Storage Multipathing software is an associated application that is bundled with the product name.

Adding SAN Devices

Adding and removing SAN devices requires knowledge of the following commands:

■ `luxadm`(1M)
■ `format`(1M)
■ `fsck`(1M)
■ `newfs`(1M)
■ `cfgadm`(1M) and `cfgadm_fp`(1M)

Note – If you use the `format` command when Solaris FC and Storage Multipathing is enabled, you will see only one instance of a device identifier for each LUN. Without the Solaris FC and Storage Multipathing, you will see one identifier for each path.

The `cfgadm` and `cfgadm_fp` commands are used most frequently to configure storage devices on a SAN. Refer to the appropriate man page for detailed instructions about how to use each command.

▼ To Add a SAN Device

1. **Create the LUN(s) desired on the SAN device.**

2. **If necessary, apply LUN masking for HBA control on the SAN device.**

3. **Connect the storage device to the system.**

4. **If necessary, create port-based or WWN zones on the switch on the SAN device.**

5. **Run the** `fsck` **or** `newfs` **commands on the device, if used for file systems.**

6. **Mount any existing file systems available on the storage device's LUNs or disk groups.**

Note – You might need to run the `fsck` command to repair any errors in the LUNs listed in the `/etc/vfstab` file.

Configuring Fabric Boot Devices on SPARC

The Solaris FC and Storage Multipathing software allows a Solaris 10 SPARC server to be booted from a fabric disk device.

Boot Considerations

Fabric boot devices can be added via the Solaris 10 interactive installation GUI or text installer just as internal boot devices have been added in the previous Solaris OS releases. Please refer to the Solaris 10 install guide for details. This feature is included on the SPARC platform only. You should consider the following points while enabling a fabric boot device.

- Minimize interference to boot device by:
  - Ensuring the boot device is not an overly subscribed target or LUN.
  - Avoiding installation of applications and software on target or LUN.
  - Reducing physical distance between host and fabric device, as well as the number of hops.
- Remove boot disk from volume manager control prior to beginning fabric boot procedure.
- Ensuring the latest HBA Fcode and drivers are loaded for the install HBA's on the system.
- Whether multipathing is desired on the boot device. This is handled by the `stmsboot` commands as described in Chapter 3 "Configuring Multipathing Software" on page 9.

Configuring IPFC SAN Devices

Configuring IPFC on the host system, describes host recognition of IPFC devices and implementation of IP over FC in a SAN. The IPFC driver is based on RFC 2625 and it allows IP traffic to run over FC.

This chapter contains the following topics:

- "IPFC Considerations" on page 29
- "Invoking and Configuring IPFC" on page 33

IPFC Considerations

The following table shows the supported features available for IPFC.

TABLE 5-1   IPFC (NFS/NAS and SNDR)

Feature	Supported
Cascading	Yes, with fabric zones only
Zone type	Fabric zone (with the HBA configured as an F-port point-to-point connection)
Maximum number of device ports per zone	253

The following restrictions apply:

- IPFC is not supported on 1 Gbit Sun switches.
- Promiscuous mode is not supported. The snoop(1M) utility cannot be used.
- Multicasting is supported through broadcasting only.
- Network cards using IPFC cannot be used as routers. The /etc/notrouter file must be present on the host.

- Any standard network commands can be used after IPFC is attached. There are not any usage differences when these commands (`telnet`, `ping`, or `ftp`) are used in an Ethernet setup.

Determining Fibre Channel Adapter Port Instances

This section explains how to configure the desired host system for IPFC. It includes the procedures to determine the port instance and to plumb an IPFC instance.

▼ To Determine Port Instances

1. **Determine the HBA PCI adapter slot and the I/O board PCI slot.**

   You need this information to perform the calculation in Step 2.

   For example, assume you have an array with an HBA card located in PCI adapter slot 5, and the PCI adapter is in slot 1 of the I/O board.

2. **Determine the instance number.**

   a. **Use an editor to search for the `fp` driver binding name in the `/etc/path_to_inst` file.**

   Entries have `fp` on the line.

   Note – Determine the correct entry by finding the hardware path described in your server hardware manual or Sun System Handbook. The Sun System Handbook is available at `http://sunsolve.sun.com/handbook_pub/`.

   ---

   b. **Narrow the search by using the I/O board and slot information from Step 1.**

   Note – The following method of deriving the Solaris 10 device path of an HBA from its physical location in server may not work for all Sun server hardware.

   ---

   i. **Multiply the PCI adapter slot number by the number of adapter ports.**

   For example, if the HBA has two ports, multiply by 2. Using the array with an HBA in the PCI adapter slot 5, multiply 5 by 2 to get 10.

   ii. **Add the PCI adapter I/O board slot number to the number derived in Step i.**

   Using an HBA in PCI adapter slot 5 and PCI slot 1 of the I/O board, add 1 to 10 for a sum of 11.

iii. Convert the number derived in Step ii to hexadecimal.

The number 11 converts to b in hexadecimal.

iv. Search for the `fp` entry with `pci@`*hex* where *hex* is the number you derived in Step iii.

The following Table 1-2, shows a single FC network adapter device path and describes the elements of the device path.

PCI Single FC Network Adapter Device Path.

```
"/pci@b,2000/SUNW,qlc@2/fp@0,0" 7 "fp"
```

TABLE 5-2   PCI Single FC Network Adapter `/etc/path_to_inst` Device Path Entry

Entry Item	Entry Value
Physical Name	`/pci@b,2000/sunw,qlc@2/fp@0,0`
Instance Number	7
Driver Binding Name	`fp`

3. Manually plumb each FP instance.

Use the `ifconfig` *<interface_number>* `plumb` command. In this example, the value of *<interface_number>* is `fcip7`.

```
# ifconfig fcip7 plumb
```

When the command is successful, a message appears on both the console and in the messages file. For example:

```
Sep 13 15:52:30 bytownite ip: ip: joining multicasts failed (7) on
fcip0 - will use link layer broadcasts for multicast
```

▼ To Plumb an IPFC Instance

Each FP instance on the system has an entry in `/dev/fc`. If HBAs have been removed, some stale links might exist. Use this procedure to load and plumb IPFC.

1. **For each entry in** `/dev/fc`**, issue a** `luxadm -e dump_map` **command to view all the devices that are visible through that HBA port:**

```
# luxadm -e dump_map /dev/fc/fp0
Pos   Port_ID Hard_Addr Port WWN          Node WWN          Type
0     610100  0         210000e08b049f53 200000e08b049f53 0x1f
(Unknown Type)
1     620d02  0         210000e08b02c32a 200000e08b02c32a 0x1f
(Unknown Type)
2     620f00  0         210000e08b03eb4b 200000e08b03eb4b 0x1f
(Unknown Type)
3     620e00  0         210100e08b220713 200100e08b220713 0x1f
(Unknown Type,Host Bus Adapter)
# luxadm -e dump_map /dev/fc/fp1
 No FC devices found. - /dev/fc/fp1
```

2. **Based on the list of devices, determine which destination HBAs are visible to the remote host with which you want to establish IPFC communications.**

   In the example for this procedure, the destination HBAs have port IDs `610100` and `620d02`. The originating HBA's port ID is `620e00`.

3. **List the physical path of the originating HBA port from which you can see the destination HBA port, where** *originating-hba-link* **is a variable for the link determined in Step 2.**

```
# ls -l /dev/fc/fp originating-hba-link
```

   For example, here **0** is the number for the *originating-hba-link*:

```
# ls -l /dev/fc/fp0
lrwxrwxrwx  1 root    root        51 Sep  4 08:23 /dev/fc/fp0 ->
../../devices/pci@8,600000/SUNW,qlc@1/fp@0,0:devctl
```

4. **Search the physical path identified in Step 3.**

   You must remove the leading `../../devices` from the path name output. For example

```
# grep pci@8,600000/SUNW,qlc@1/fp@0,0 /etc/path_to_inst
"/pci@8,600000/SUNW,qlc@1/fp@0,0" 0 "fp"
```

5. **Determine the fp instance for the originating HBA port from the output of the command in Step 4.**

   The instance number precedes "`fp`" in the output. In the following example output, the instance number is `0`.

   ```
   "/pci@8,600000/SUNW,qlc@1/fp@0,0" 0 "fp"
   ```

6. **Use the instance number from Step 5 to load IPFC and plumb the IPFC interface.**

   In this example, the instance is 0.

   ```
   # ifconfig fcip0 plumb
   ```

Invoking and Configuring IPFC

Immediately upon installation, start IPFC manually with the `ifconfig` command. You can configure the host so that on subsequent reboot, the IPFC network interface starts automatically. This section describes the procedures to start a network interface manually and to configure the host for automatic plumbing upon reboot.

▼ To Start a Network Interface Manually

Use this procedure when you want to plumb IPFC with specific netmask values and get the IPFC interface up and running.

1. **Use the `ifconfig` command with the appropriate interface.**

   Ask your network administrator for an appropriate IP address and netmask information. For example, to enable an IPFC interface associated with fp instance `0` and an IP address of `192.9.201.10`, type:

   ```
   # touch /etc/notrouter
   # ifconfig fcip0 inet 192.9.201.10 netmask 255.255.255.0 up
   ```

   The `ifconfig` command is described in more detail in the `ifconfig`(1M) manpage.

2. **Use the command** `ifconfig -a` **to verify the network is functioning.**

   The output of `ifconfig -a` should look like this:

```
lo0: flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4> mtu 8232
index 1
inet 127.0.0.1 netmask ff000000
fcip0: flags=
1001843<UP,BROADCAST,RUNNING,MULTICAST,MULTI_BCAST,IPv4> mtu 1500
index 2
inet 192.9.201.10 netmask ffffff00 broadcast 192.9.201.255 ether
0:e0:8b:1:3c:f7
hme0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500
index 3
inet 192.9.200.70 netmask ffffff00 broadcast 192.9.200.255
ether 8:0:20:fc:e9:49
```

▼ To Configure the Host for Automatic Plumbing Upon Reboot

Each network interface must have an `/etc/hostname.interface` file defining the name of the IP address associated with it. For example, IPFC network interface `fcip0` has a file named `/etc/hostname.fcip0`.

1. **Manually create a** `/etc/hostname.`*interface* **file with a text editor so it contains a single line that identifies the host name or interface IP address.**

2. **Use a text editor to make any additional entries to the** `/etc/inet/hosts` **file.**

   The Solaris 10 installation program creates the `/etc/inet/hosts` file with minimum entries. You must manually make additional entries with a text editor. (See the `hosts(4)` man page for additional information.)

   The `/etc/inet/hosts` file contains the hosts database. This file contains the host names and the primary network interface IP addresses, as well as the IP addresses of other network interfaces attached to the system and of any other network interfaces that the machine must know about.

   The following code shows an example of an `etc/inet/host` file.

   CODE EXAMPLE 5-1 `sun1 machine etc/inet/hosts`

```
127.0.0.1     localhost     loghost
192.9.200.70  sun1     #This is the local host name
192.9.201.10 fcip0 #Interface to network 192.9.201.10
```

3. **Edit the `/etc/nsswitch.conf` file so that all un-commented entries have the word `files` before any other nameservice.**

The `/etc/nsswitch.conf` specifies which name service to use for a particular machine. The following code shows an example of an `/etc/nsswitch.conf` file.

CODE EXAMPLE 5-2 sun1 machine /etc/nsswitch.conf File

```
hosts: files nis
```

Manual Configuration for Fabric Connected Devices

This appendix explains about configuring and unconfiguring the fabric devices in Solaris 10. It explains how the visible fabric devices on a host are detected and configured without enabling the multipathing feature and enabling the multipathing software.

This appendix contains the following sections:

Manual Configuration

In Solaris 10, fabric-connected devices are made available to the Solaris host. This differs from previous versions of the Solaris FC and Storage Multipathing software running on the Solaris 8 or the Solaris 9 OS. In those versions, manual configuration steps were required to achieve fabric-connected device availability on the Solaris 10 host. If you want to manually configure the fabric-connected devices, use the following steps to change the default behavior on the Solaris 10 host. Note that changing the default behavior makes all of your fabric-connected devices unavailable. This can cause problem for fabric-connected devices that are required to be available at boot time.

1. **Edit the file** `/kernel/drv/fp.conf` **and uncomment the line.**

```
# manual_configuration_only=1;
```

By removing the leading '#

```
manual_configuration_only=1;
```

Please refer to the `fp`(7D) man page and the `cfgadm_fp`(1M) man page for further information on this setting.

2. **Reboot the host.**

3. **For each fabric-connected device that will need to be made available, follow the steps in "Configuring Device Nodes Without Multipathing Enabled" on page 41 or "Configuring Device Nodes With Solaris Multipathing Software Enabled" on page 45 dependent upon whether you are using the Solaris FC and Storage Multipathing Software for multipathing.**

If the original default behavior for fabric-connected devices is desired, use the following steps:

4. **Edit the file** "`/kernel/drv/fp.conf`" **and comment the line**

```
manual_configuration_only=1;
```

By adding a leading '#'

```
# manual_configuration_only=1;
```

Please refer to the `fp`(7D) man page and the `cfgadm_fp`(1M) man page for further information on this setting.

5. **Reboot the host**

Fabric Device Node Configuration

After you configure the hardware in your direct-attach system or SAN, you must ensure that the hosts recognize the devices. This section explains host recognition of fabric devices, also known as 24-bit FC addressing devices on the SAN. After

configuring the devices, ports, and zones in your SAN, you need to make sure that the host is aware of the devices. You can have up to 16 million fabric devices connected together on a SAN with FC support.

This section is limited to the operations required from the perspective of the Solaris 10 OS. It does not cover other aspects, such as device availability and device-specific management. If devices are managed by other software, such as a volume manager, refer to the volume manager product documentation for additional instructions.

Ensuring That LUN Level Information is Visible

You need to follow the steps mentioned below to ensure the visibility of LUN level information.

1. **Use the** `cfgadm -al -o show_SCSI_LUN` *<controller_id>* **command to identify LUN level information.**

   If you issue the `cfgadm -al -o show_SCSI_LUN` *<controller_id>* command immediately after a system boots up, the output might not show the Fibre Channel Protocol (FCP) SCSI LUN level information. The information does not appear because the storage device drivers, such as the `ssd` and `st` driver, are not loaded on the running system.

2. **Use the** `modinfo` **command to check whether the drivers are loaded. After the drivers are loaded, the LUN level information is visible in the** `cfgadm` **output.**

▼ To Detect Fabric Devices Visible on a Host

This section provides an example of the procedure for detecting fabric devices using FC host ports `c0` and `c1`. This procedure also shows the device configuration information that is displayed with the `cfgadm`(1M) command.

Note – In the following examples, only failover path attachment point IDs (Ap_Ids) are listed. The Ap_Ids displayed on your system depend on your system configuration.

1. **Become superuser.**

2. **Display the information about the attachment points on the system.**

```
# cfgadm -l
Ap_Id              Type        Receptacle  Occupant    Condition
c0                 fc-fabric   connected   unconfigured unknown
c1                 fc-private  connected    configured   unknown
```

In this example, `c0` represents a fabric-connected host port, and `c1` represents a private, loop-connected host port. Use the `cfgadm(1M)` command to manage the device configuration on fabric-connected host ports.

By default, the device configuration on private, loop-connected host ports is managed by a host using the Solaris 10 OS.

3. **Display information about the host ports and their attached devices.**

```
# cfgadm -al
Ap_Id                   Type        Receptacle   Occupant    Condition
c0                      fc-fabric   connected    unconfigured unknown
c0::50020f2300006077 disk          connected    unconfigured unknown
c0::50020f23000063a9 disk          connected    unconfigured unknown
c0::50020f2300005f24 disk          connected    unconfigured unknown
c0::50020f2300006107 disk          connected    unconfigured unknown
c1                      fc-private  connected    configured   unknown
c1::220203708b69c32b disk          connected    configured   unknown
c1::220203708ba7d832 disk          connected    configured   unknown
c1::220203708b8d45f2 disk          connected    configured   unknown
c1::220203708b9b20b2 disk          connected    configured   unknown
```

Note – The `cfgadm -l` command displays information about FC host ports. Also use the `cfgadm -al` command to display information about FC devices. The lines that include a port world wide name (WWN) in the `Ap_Id` field associated with `c0` represent a fabric device. Use the `cfgadm configure` and `unconfigure` commands to manage those devices and make them available to hosts using the Solaris 10 OS. The Ap_Id devices with port WWNs under `c1` represent private-loop devices that are configured through the `c1` host port.

Configuring Device Nodes Without Multipathing Enabled

This section describes fabric device configuration tasks on a host that does not have multipathing enabled.

The procedures in this section use specific devices as examples to illustrate how to use the cfgadm(1M) command to detect and configure fabric devices.

The procedures in this section show how to detect fabric devices that are visible on a host and to configure and make them available to a host using the Solaris 10 OS.

The device information that you supply and that is displayed by the cfgadm(1M) command depends on your system configuration.

▼ To Configure an Individual Device Without Multipathing

This sample procedure describes how to configure a fabric device that is attached to the fabric-connected host port c0.

1. **Become superuser.**

2. **Identify the device to be configured.**

```
# cfgadm -al
Ap_Id                   Type       Receptacle   Occupant       Condition
c0                      fc-fabric  connected    unconfigured unknown
c0::50020f2300006077 disk         connected    unconfigured unknown
c0::50020f23000063a9 disk         connected    unconfigured unknown
c0::50020f2300005f24 disk         connected    unconfigured unknown
c0::50020f2300006107 disk         connected    unconfigured unknown
c1                      fc-private connected    configured     unknown
c1::220203708b69c32b disk         connected    configured     unknown
c1::220203708ba7d832 disk         connected    configured     unknown
c1::220203708b8d45f2 disk         connected    configured     unknown
c1::220203708b9b20b2 disk         connected    configured     unknown
```

3. **Configure the fabric device.**

```
# cfgadm -c configure c0::50020f2300006077
```

4. **Verify that the selected fabric device is configured.**

```
# cfgadm -al
Ap_Id                   Type        Receptacle  Occupant     Condition
c0                      fc-fabric   connected   configured   unknown
c0::50020f2300006077 disk          connected   configured   unknown
c0::50020f23000063a9 disk          connected   unconfigured unknown
c0::50020f2300005f24 disk          connected   unconfigured unknown
c0::50020f2300006107 disk          connected   unconfigured unknown
c1                      fc-private  connected   configured   unknown
c1::220203708b69c32b disk          connected   configured   unknown
c1::220203708ba7d832 disk          connected   configured   unknown
c1::220203708b8d45f2 disk          connected   configured   unknown
c1::220203708b9b20b2 disk          connected   configured   unknown
```

Notice that the Occupant column for both c0 and c0::50020f2300006077 displays as configured, indicating that the c0 port has a configured occupant and that the c0::50020f2300006077 device is configured.

5. **Use the show_SCSI_LUN option to display FCP SCSI LUN information for multi-LUN SCSI devices.**

The following code example shows that the physical devices connected through Ap_Id c0:50020f2300006077 have four LUNs configured.

CODE EXAMPLE A-1 shows cfgadm output for four LUN devices

```
# cfgadm -al -o show_SCSI_LUN c0
Ap_Id                   Type        Receptacle  Occupant     Condition
c0                      fc-fabric   connected   configured   unknown
c0::50020f2300006077,0 disk        connected   configured   unknown
c0::50020f2300006077,1 disk        connected   configured   unknown
c0::50020f2300006077,2 disk        connected   configured   unknown
c0::50020f2300006077,3 disk        connected   configured   unknown
```

The device is now available on the host using the Solaris 10 OS. The paths represent each SCSI LUN in the physical device represented by c0::50020f2300006077.

▼ To Configure Multiple Devices Without Multipathing

Make sure you first identify the devices visible to the host with the procedure "To Detect Fabric Devices Visible on a Host" on page 39. This procedure describes how to configure all unconfigured fabric devices that are attached to a fabric-connected host port. The port used as an example is c0.

1. **Become superuser.**

2. **Identify the devices to be configured.**

```
# cfgadm -al
Ap_Id                   Type         Receptacle   Occupant     Condition
c0                      fc-fabric    connected    unconfigured unknown
c0::50020f2300006077 disk           connected    unconfigured unknown
c0::50020f23000063a9 disk           connected    unconfigured unknown
c0::50020f2300005f24 disk           connected    unconfigured unknown
c0::50020f2300006107 disk           connected    unconfigured unknown
c1                      fc-private   connected    configured   unknown
c1::220203708b69c32b disk           connected    configured   unknown
c1::220203708ba7d832 disk           connected    configured   unknown
c1::220203708b8d45f2 disk           connected    configured   unknown
c1::220203708b9b20b2 disk           connected    configured   unknown
```

3. **Configure all of the unconfigured devices on the selected port.**

```
# cfgadm -c configure c0
```

Note – This operation repeats the configure operation of an individual device for all the devices on c0. This can be time consuming if the number of devices on c0 is large.

4. **Verify that all devices on `c0` are configured.**

```
# cfgadm -al
Ap_Id                 Type        Receptacle  Occupant    Condition
c0                    fc-fabric   connected   configured  unknown
c0::50020f2300006077 disk        connected   configured  unknown
c0::50020f23000063a9 disk        connected   configured  unknown
c0::50020f2300005f24 disk        connected   configured  unknown
c0::50020f2300006107 disk        connected   configured  unknown
c1                    fc-private  connected   configured  unknown
c1::220203708b69c32b disk        connected   configured  unknown
c1::220203708ba7d832 disk        connected   configured  unknown
c1::220203708b8d45f2 disk        connected   configured  unknown
c1::220203708b9b20b2 disk        connected   configured  unknown
```

5. **Use the `show_SCSI_LUN` command to display FCP SCSI LUN information for multi-LUN SCSI devices.**

   The following code example shows that the physical devices represented by `c0::50020f2300006077` and `c0::50020f2300006107` each have four LUNs configured. The physical devices represented by `c0::50020f23000063a9` and `c0::50020f2300005f24` each have two LUNs configured.

   CODE EXAMPLE A-2 `show_SCSI_LUN` Output for Multiple LUNs and Two Devices

```
# cfgadm -al -o show_SCSI_LUN c0
Ap_Id                   Type    Receptacle   Occupant     Condition
c0                      fc-fabric connected   configured   unknown
c0::50020f2300006077,0 disk    connected   configured   unknown
c0::50020f2300006077,1 disk    connected   configured   unknown
c0::50020f2300006077,2 disk    connected   configured   unknown
c0::50020f2300006077,3 disk    connected   configured   unknown
c0::50020f23000063a9,0 disk    connected   configured   unknown
c0::50020f23000063a9,1 disk    connected   configured   unknown
c0::50020f2300005f24,0 disk    connected   configured   unknown
c0::50020f2300005f24,1 disk    connected   configured   unknown
c0::50020f2300006107,0 disk    connected   configured   unknown
c0::50020f2300006107,1 disk    connected   configured   unknown
c0::50020f2300006107,2 disk    connected   configured   unknown
c0::50020f2300006107,3 disk    connected   configured   unknown
```

Configuring Device Nodes With Solaris Multipathing Software Enabled

This section describes how to perform fabric device configuration steps on a host that has the Solaris Multipathing software enabled.

The devices attached to the fabric-connected host port are not configured by default and so are not available to the host using the Solaris 10 OS. Use the `cfgadm`(1M) `configure` and `unconfigure` commands to manage device node creation for fabric devices. See the `cfgadm_fp`(1M) man page for additional information. The procedures in this section illustrate steps to detect fabric devices that are visible on a host and to configure them as multipathing devices to make them available to the host using the Solaris 10 OS.

The device information that you supply, and that is displayed by the `cfgadm`(1M) command, depends on your system configuration. For more information on the `cfgadm` command, see the `cfgadm_fp`(1M) and `cfgadm`(1M) man pages.

▼ To Configure Individual Devices With Solaris Multipathing Software

This sample procedure uses fabric-connected host ports `c0` and `c2` to configure fabric devices as multipath devices on a host that has the Solaris FC and Storage Multipathing software enabled.

The `cfgadm -c configure` command for fabric devices is the same regardless of whether or not multipathing is enabled.

1. Become superuser.

2. **Identify the port WWN of the device to be configured as the Solaris FC and Storage Multipathing software device.**

   Look for devices on a fabric-connected host port, marked as `fc-fabric`. These are the devices you can configure with the `cfgadm -c configure` command.

   CODE EXAMPLE A-3 `cfgadm` Listing of Fabric and Private-Loop Devices

   ```
   # cfgadm -al
   Ap_Id                   Type        Receptacle  Occupant     Condition
   c0                      fc-fabric   connected    unconfigured unknown
   c0::50020f2300006077 disk          connected    unconfigured unknown
   c0::50020f23000063a9 disk          connected    unconfigured unknown
   c1                      fc-private  connected    configured   unknown
   c1::220203708b69c32b disk          connected    configured   unknown
   c1::220203708ba7d832 disk          connected    configured   unknown
   c1::220203708b8d45f2 disk          connected    configured   unknown
   c1::220203708b9b20b2 disk          connected    configured   unknown
   c2                      fc-fabric   connected    unconfigured unknown
   c2::50020f2300005f24 disk          connected    unconfigured unknown
   c2::50020f2300006107 disk          connected    unconfigured unknown
   ```

   In Code example A-3, the `c0::50020f2300006077` and `c2::50020f2300006107` `Ap_Ids` represent the same storage device with different port WWNs for the storage device controllers. The `c0` and `c2` host ports are enabled for use by the Solaris FC and Storage Multipathing software.

3. **Configure the fabric device and make the Solaris FC and Storage Multipathing software devices available to the host.**

   ```
   # cfgadm -c configure c0::50020f2300006077 c2::50020f2300006107
   ```

4. Verify that the selected devices are configured.

```
# cfgadm -al
Ap_Id                   Type        Receptacle  Occupant      Condition
c0                      fc-fabric   connected   configured    unknown
c0::50020f2300006077 disk          connected   configured    unknown
c0::50020f23000063a9 disk          connected   unconfigured unknown
c1                      fc-private  connected   configured    unknown
c1::220203708b69c32b disk          connected   configured    unknown
c1::220203708ba7d832 disk          connected   configured    unknown
c1::220203708b8d45f2 disk          connected   configured    unknown
c1::220203708b9b20b2 disk          connected   configured    unknown
c2                      fc-fabric   connected   configured    unknown
c2::50020f2300005f24 disk          connected   unconfigured unknown
c2::50020f2300006107 disk          connected   configured    unknown
```

Notice that the `Occupant` column of `c0` and `c0::50020f2300006077` specifies `configured`, which indicates that the `c0` port has at least one configured occupant and that the `c0::50020f2300006077` device is configured. The same change has been made in `c2` and `c2::50020f2300006107`.

When the configure operation has been completed without an error, multipathing-enabled devices are created on the host. If the physical device represented by `c0::50020f2300006077` and `c2::50020f2300006107` has multiple SCSI LUNs configured, each LUN is configured as a Solaris FC and Storage Multipathing software device. Code example A-4, shows that two LUNs are configured through `c0::50020f2300006077` and `c2::50020f2300006107`. Each `Ap_Id` is associated with a path to those Solaris FC and Storage Multipathing software devices.

CODE EXAMPLE A-4   show_SCSI_LUN Output for Two LUNs on a Device

```
# cfgadm -al -o show_SCSI_LUN c0::50020f2300006077\
c2::50020f2300006107
Ap_Id                     Type      Receptacle  Occupant      Condition
c0::50020f2300006077,0 disk         connected   configured    unknown
c0::50020f2300006077,1 disk         connected   configured    unknown
c2::50020f2300006107,0 disk         connected   configured    unknown
c2::50020f2300006107,1 disk         connected   configured    unknown
```

In code example A-4, notice that the following two multipathing-enabled devices are created for the device represented by `c0::50020f2300006077` and `c2::50020f2300006107`:

▼ To Configure Multiple Devices With Multipathing

Before you configure or remove device nodes, be sure to first identify the fabric devices by using the procedure "To Detect Fabric Devices Visible on a Host" on page 39.

In this example, an Ap_Id on a fabric-connected host port is a path to the Solaris FC and Storage Multipathing software device. For example, all devices with a path through `c2` are to be configured, but none through `c0` are to be configured. `c2` is an attachment point from the host to the fabric, whereas `c2::50020f2300006107` is an attachment point from the storage to the fabric. A host detects all the storage devices in a fabric for which it is configured.

Configuring an Ap_Id on the Solaris FC and Storage Multipathing software device that has already been configured through another Ap_Id results in an additional path to the previously configured device. Note that a new device node is not created in this case. The device node is created only the first time an Ap_Id to the corresponding Solaris FC and Storage Multipathing device is configured.

1. **Become superuser.**

2. **Identify the fabric-connected host port to be configured.**

```
# cfgadm -al
Ap_Id                   Type          Receptacle  Occupant     Condition
c0                      fc-fabric     connected   configured   unknown
c0::50020f2300006077 disk            connected   configured   unknown
c0::50020f23000063a9 disk            connected   configured   unknown
c1                      fc-private    connected   configured   unknown
c1::220203708b69c32b disk            connected   configured   unknown
c1::220203708ba7d832 disk            connected   configured   unknown
c1::220203708b8d45f2 disk            connected   configured   unknown
c1::220203708b9b20b2 disk            connected   configured   unknown
c2                      fc-fabric     connected   unconfigured unknown
c2::50020f2300005f24 disk            connected   unconfigured unknown
c2::50020f2300006107 disk            connected   unconfigured unknown
```

Devices represented by Ap_Ids `c0::50020f2300006077` and `c2::50020f2300006107` are two paths to the same physical device, with `c0::50020f2300006077` already configured. Configure the unconfigured devices on the selected port. This operation repeats the `configure` command of an individual device for all the devices on `c2`. This can be time-consuming if the number of devices on `c2` is large.

```
# cfgadm -c configure c2
```

3. **Verify that all devices on** `c2` **are configured.**

```
# cfgadm -al
Ap_Id                  Type        Receptacle  Occupant    Condition
c0                     fc-fabric   connected    configured   unknown
c0::50020f2300006077 disk         connected    configured   unknown
c0::50020f23000063a9 disk         connected    configured   unknown
c1                     fc-private  connected    configured   unknown
c1::220203708b69c32b disk         connected    configured   unknown
c1::220203708ba7d832 disk         connected    configured   unknown
c1::220203708b8d45f2 disk         connected    configured   unknown
c1::220203708b9b20b2 disk         connected    configured   unknown
c2                     fc-fabric   connected    configured   unknown
c2::50020f2300005f24 disk         connected    configured   unknown
c2::50020f2300006107 disk         connected    configured   unknown
```

Notice that the Occupant column of c2 and all of the devices under c2 is marked as configured.

The show_SCSI_LUN command displays FCP SCSI LUN information for multiple LUN SCSI devices. Code example A-5, shows that the physical devices connected through by `c2::50020f2300006107` and `c2::50020f2300005f24` each have two LUNs configured.

CODE EXAMPLE A-5   show_SCSI_LUN Output for Multiple LUNs and Multiple Physical Devices

```
# cfgadm -al -o show_SCSI_LUN c2
Ap_Id                    Type        Receptacle  Occupant    Condition
c2                       fc-fabric connected    configured   unknown
c2::50020f2300005f24,0 disk       connected    configured   unknown
c2::50020f2300005f24,1 disk       connected    configured   unknown
c2::50020f2300006107,0 disk       connected    configured   unknown
c2::50020f2300006107,1 disk       connected    configured   unknown
```

Unconfiguring Fabric Devices

This section provides information about unconfiguring the Solaris FC and Storage Multipathing software.

Unconfiguring a Fabric Device

Before you unconfigure a fabric device, stop all activity to the device and unmount any file systems on the fabric device. See the administration documentation for the Solaris 10 OS for unmounting instructions. If the device is under any volume manager's control, see the documentation for your volume manager before unconfiguring the device.

▼ To Unconfigure a Fabric Device

This procedure describes how to unconfigure a fabric device that is attached to the fabric-connected host port c0.

1. **Become superuser.**

2. **Identify the device to be unconfigured.**

   Only devices on a fabric-connected host port can be unconfigured.

   ```
   # cfgadm -al
   Ap_Id                   Type        Receptacle   Occupant    Condition
   c0                      fc-fabric   connected    configured   unknown
   c0::50020f2300006077 disk          connected    configured   unknown
   c0::50020f23000063a9 disk          connected    configured   unknown
   c1                      fc-private  connected    configured   unknown
   c1::220203708b69c32b disk          connected    configured   unknown
   c1::220203708ba7d832 disk          connected    configured   unknown
   ```

3. **Unconfigure the fabric device.**

   ```
   # cfgadm -c unconfigure c0::50020f2300006077
   ```

4. **Verify that the selected fabric device is unconfigured.**

   ```
   # cfgadm -al
   Ap_Id                   Type        Receptacle   Occupant      Condition
   c0                      fc-fabric   connected    configured     unknown
   c0::50020f2300006077 disk          connected    unconfigured  unknown
   c0::50020f23000063a9 disk          connected    configured     unknown
   c1                      fc-private  connected    configured     unknown
   c1::220203708b69c32b disk          connected    configured     unknown
   c1::220203708ba7d832 disk          connected    configured     unknown
   ```

▼ To Unconfigure all Fabric Devices on a Fabric-Connected Host Port

This procedure describes how to unconfigure all configured fabric devices that are attached to a fabric-connected host port.

1. **Become superuser.**

2. **Identify the fabric devices to be unconfigured.**

   Only devices on a fabric-connected host port can be unconfigured.

   ```
   # cfgadm -al
   Ap_Id                 Type         Receptacle   Occupant    Condition
   c0                    fc-fabric    connected    configured  unknown
   c0::50020f2300006077 disk         connected    configured  unknown
   c0::50020f23000063a9 disk         connected    configured  unknown
   c1                    fc-private   connected    configured  unknown
   c1::220203708b69c32b disk         connected    configured  unknown
   c1::220203708ba7d832 disk         connected    configured  unknown
   ```

3. **Stop all activity to each fabric device on the selected port and unmount any file systems on each fabric device. If the device is under any volume manager's control, see the documentation for your volume manager before unconfiguring the device.**

   ```
   # cfgadm -c unconfigure c0
   ```

4. **Unconfigure all of the configured fabric devices on a selected port.**

Note – This operation repeats the unconfigure operation of an individual device for all the devices on c0. This can be time-consuming if the number of devices on c0 is large.

---

5. **Verify that all the devices on `c0` are unconfigured.**

```
# cfgadm -al
Ap_Id                 Type          Receptacle   Occupant     Condition
c0                    fc-fabric     connected    unconfigured unknown
c0::50020f2300006077 disk          connected    unconfigured unknown
c0::50020f23000063a9 disk          connected    unconfigured unknown
c1                    fc-private    connected    configured   unknown
c1::220203708b69c32b disk          connected    configured   unknown
c1::220203708ba7d832 disk          connected    configured   unknown
```

Notice that the `Occupant` column of c0 and all the fabric devices attached to it are displayed as `unconfigured`.

▼ To Unconfigure a Fabric Device Associated With Solaris FC and Storage Multipathing Enabled Devices

This procedure shows fabric-connected host ports `c0` and `c2` to illustrate how to unconfigure fabric devices associated with Solaris FC and Storage Multipathing software devices.

1. **Become superuser.**

2. **Identify the port WWN of the fabric device to be unconfigured.**

```
# cfgadm -al
Ap_Id                 Type          Receptacle   Occupant     Condition
c0                    fc-fabric     connected    configured   unknown
c0::50020f2300006077 disk          connected    configured   unknown
c0::50020f23000063a9 disk          connected    configured   unknown
c1                    fc-private    connected    configured   unknown
c1::220203708b69c32b disk          connected    configured   unknown
c1::220203708ba7d832 disk          connected    configured   unknown
c2                    fc-fabric     connected    configured   unknown
c2::50020f2300005f24 disk          connected    configured   unknown
c2::50020f2300006107 disk          connected    configured   unknown
```

In this example, the `c0::50020f2300006077` and `c2::50020f2300006107` Ap_Ids represent different port WWNs for the same device associated with a Solaris FC and Storage Multipathing software device. The `c0` and `c2` host ports are enabled for use by the Solaris FC and Storage Multipathing software.

3. **Stop all device activity to each fabric device on the selected port and unmount any file systems on each fabric device. If the device is under any volume manager's control, see the documentation for your volume manager for maintaining the fabric device.**

4. **Unconfigure fabric devices associated with the Solaris FC and Storage Multipathing software device.**

   Only devices on a fabric-connected host port can be unconfigured through the `cfgadm -c unconfigure` command.

   ```
   # cfgadm -c unconfigure c0::50020f2300006077 c2::50020f2300006107
   ```

   Note – You can remove a device from up to eight paths individually, as in the example command `cfgadm -c unconfigure c0::1111, c1::2222, c3::3333`, etc. As an alternative, you can remove an entire set of paths from the host, as in the example `cfgadm -c unconfigure c0`.

5. **Verify that the selected devices are unconfigured.**

   ```
   # cfgadm -al
   Ap_Id                 Type        Receptacle   Occupant     Condition
   c0                    fc-fabric   connected    configured   unknown
   c0::50020f2300006077 disk        connected    unconfigured unknown
   c0::50020f23000063a9 disk        connected    configured   unknown
   c1                    fc-private  connected    configured   unknown
   c1::220203708b69c32b disk        connected    configured   unknown
   c1::220203708ba7d832 disk        connected    configured   unknown
   c2                    fc-fabric   connected    configured   unknown
   c2::50020f2300005f24 disk        connected    configured   unknown
   c2::50020f2300006107 disk        connected    unconfigured unknown
   ```

   Notice that the Ap_Ids `c0::50020f2300006077` and `c2::50020f2300006107` are unconfigured. The `Occupant` column of `c0` and `c2` still displays those ports as `configured` because they have other configured occupants.

   The Solaris FC and Storage Multipathing software devices associated with the Ap_Ids `c0::50020f2300006077` and `c2::50020f2300006107` are no longer available to the host using the Solaris 10 OS. The following two Solaris FC and Storage Multipathing devices are removed from the host:

   /dev/rdsk/c6t60020F20000061073AC8B52D000B74A3d0s2

   /dev/rdsk/c6t60020F20000061073AC8B4C50004ED3Ad0s2

▼ To Unconfigure One Path to a Multipathing Device

In contrast to the procedure in the preceding section, this procedure shows how to unconfigure one device associated with `c2::50020f2300006107` and leave the other device, `50020f2300006077`, configured. Only devices on a fabric-connected host port can be unconfigured through the `cfgadm unconfigure` command.

1. **Become superuser.**

2. **Identify the Ap_Id of the Solaris FC and Storage Multipathing software device to be unconfigured.**

```
# cfgadm -al
Ap_Id                   Type        Receptacle   Occupant    Condition
c0                      fc-fabric   connected    configured   unknown
c0::50020f2300006077 disk          connected    configured   unknown
c0::50020f23000063a9 disk          connected    configured   unknown
c1                      fc-private  connected    configured   unknown
c1::220203708b69c32b disk          connected    configured   unknown
c1::220203708ba7d832 disk          connected    configured   unknown
c2                      fc-fabric   connected    configured   unknown
c2::50020f2300005f24 disk          connected    configured   unknown
c2::50020f2300006107 disk          connected    configured   unknown
```

In this example, `c0::50020f2300006077` and `c2::50020f2300006107` Ap_Ids represent different port WWNs for the same device.

3. **Unconfigure the Ap_Id associated with Solaris FC Storage Multipathing software device.**

Note – If the Ap_Id represents the last configured path to the Solaris FC and Storage Multipathing software device, stop all activity to the path and unmount any file systems on it. If the multipathing device is under any volume manager's control, see the documentation for your volume manager for maintaining the fabric device.

In the example that follows, the path represented as `c2::50020f2300006107` is unconfigured, and `c0::50020f2300006077` remains configured to show how you can unconfigure just one of multiple paths for a multipathing device.

```
# cfgadm -c unconfigure c2::50020f2300006107
```

4. **Verify that the selected path** `c2::50020f2300006107` **is unconfigured.**

```
# cfgadm -al
Ap_Id                 Type         Receptacle   Occupant     Condition
c0                    fc-fabric    connected    configured   unknown
c0::50020f2300006077  disk         connected    configured   unknown
c0::50020f23000063a9  disk         connected    configured   unknown
c1                    fc-private   connected    configured   unknown
c1::220203708b69c32b  disk         connected    configured   unknown
c1::220203708ba7d832  disk         connected    configured   unknown
c2                    fc-fabric    connected    configured   unknown
c2::50020f2300005f24  disk         connected    configured   unknown
c2::50020f2300006107  disk         connected    unconfigured unknown
```

The Solaris FC and Storage Multipathing software devices associated with that Ap_Id are still available to a host using the Solaris 10 OS through the other path, represented by `c0::50020f2300006077`. A device can be connected to multiple Ap_Ids and an Ap_Id can be connected to multiple devices..

```
/dev/rdsk/c6t60020F20000061073AC8B52D000B74A3d0s2

and

/dev/rdsk/c6t60020F20000061073AC8B4C50004ED3Ad0s2
```

▼ To Unconfigure all Fabric-Connected Devices With Multipathing Enabled

An Ap_Id on a fabric-connected host port is a path to the Solaris FC and Storage Multipathing software device.

When a Solaris FC and Storage Multipathing software device has multiple Ap_Ids connected to it, the device is still available to the host using the Solaris 10 OS after you unconfigure an Ap_Id. After you unconfigure the last Ap_Id, no additional paths remain and the Solaris FC and Storage Multipathing software device is unavailable to the host using the Solaris 10 OS. Only devices on a fabric-connected host port can be unconfigured.

1. **Become superuser.**

2. **Identify the devices to be unconfigured.**

```
# cfgadm -al
Ap_Id                 Type        Receptacle  Occupant    Condition
c0                    fc-fabric   connected   configured   unknown
c0::50020f2300006077 disk        connected   configured   unknown
c0::50020f23000063a9 disk        connected   configured   unknown
c1                    fc-private  connected   configured   unknown
c1::220203708b69c32b disk        connected   configured   unknown
c1::220203708b9b20b2 disk        connected   configured   unknown
c2                    fc-fabric   connected   configured   unknown
c2::50020f2300005f24 disk        connected   configured   unknown
```

3. **Unconfigure all of the configured devices on the selected port.**

```
# cfgadm -c unconfigure c2
```

Note – This operation repeats the unconfigure command of an individual device for all devices on c2. This can be time-consuming if the number of devices on c2 is large.

4. **Verify that all devices on c2 are unconfigured.**

```
# cfgadm -al
Ap_Id                 Type        Receptacle  Occupant     Condition
c0                    fc-fabric   connected   configured   unknown
c0::50020f2300006077 disk        connected   configured   unknown
c1                    fc-private  connected   configured   unknown
c1::220203708b69c32b disk        connected   configured   unknown
c1::220203708ba7d832 disk        connected   configured   unknown
c2                    fc-fabric   connected   unconfigured unknown
c2::50020f2300005f24 disk        connected   unconfigured unknown
c2::50020f2300006107 disk        connected   unconfigured unknown
```

Notice that the Occupant column lists c2 and all the devices attached to c2 as unconfigured.

Supported FC-HBA API

This appendix contains the list of supported and unsupported FC-HBA Interfaces. For further information regarding the API, refer to the Solaris FC and Storage Multipathing "Software Features" on page 2.

TABLE B-1   Supported and Un-Supported FC-HBA Interfaces

FC - HBA	Solaris FC and Storage Multipathing Software Supported
HBA_GetVersion	Yes
HBA_LoadLibrary	Yes
HBA_FreeLibrary	Yes
HBA_GetNumberofAdapters	Yes
HBA_GetAdapterName	Yes
HBA_OpenAdapter	Yes
HBA_CloseAdapter	Yes
HBA_GetAdapterAttributes	Yes
HBA_GetAdapterPortAttributes	Yes
HBA_GetDiscoveredPortAttributes	Yes
HBA_GetPortAttributesbyWWN	Yes
HBA_SendCTPassThru	Yes
HBA_SendCTPassThruV2	Yes
HBA_RefreshInformation	Yes
HBA_GetFcpTargetMapping	Yes
HBA_SendScsiInquiry	Yes
HBA_SendReportLuns	Yes
HBA_SendReadCapacity	Yes

TABLE B-1 Supported and Un-Supported FC-HBA Interfaces

FC - HBA	Solaris FC and Storage Multipathing Software Supported
HBA_GetPortStatistics	Yes
HBA_ResetStatistics	No
HBA_GetFcpPersistentBinding	No
HBA_GetEventBuffer	No
HBA_SetRNIDMgmtInfo	Yes
HBA_GetRNIDMgmtInfo	Yes
HBA_SendRNID	Yes
HBA_SendRNIDV2	Yes
HBA_ScsiInquiryV2	Yes
HBA_ScsiReportLUNsV2	Yes
HBA_ScsiReadCapacityV2	Yes
HBA_OpenAdapterByWWN	Yes
HBA_RefreshAdapterConfiguration	Yes
HBA_GetVendorLibraryAttributes	Yes
HBA_GetFcpTargetMappingV2	Yes
HBA_SendRPL	No
HBA_SendRPS	No
HBA_SendSRL	No
HBA_SendLIRR	No
HBA_SendRLS	Yes
HBA_RemoveCallback	Yes
HBA_RegisterForAdapterEvents	Yes
HBA_RegisterForAdapterAddEvents	Yes
HBA_RegisterForAdapterPortEvents	Yes
HBA_RegisterForAdapterPortStatEvents	No
HBA_RegisterForTargetEvents	Yes
HBA_RegisterForAdapterLinkEvents	No
HBA_RegisterForAdapterTargetEvents	Yes
HBA_GetFC4Statistics	No
HBA_GetFCPStatistics	No

TABLE B-1 Supported and Un-Supported FC-HBA Interfaces

FC - HBA	Solaris FC and Storage Multipathing Software Supported
HBA_GetBindingCapability	No
HBA_GetBindingSupport	No
HBA_SetBindingSupport	No
HBA_SetPersistentBindingV2	No
HBA_GetPersistentBindingV2	No
HBA_RemovePersistentBinding	No
HBA_RemoveAllPersistentBindings	No

Multipathing Troubleshooting

This appendix provides solutions to potential problems that may occur while running the Solaris FC and Storage Multipathing software.

This appendix contains the following sections:

- "System Fails to Boot During `stmsboot`" on page 61
- "System Crashes During `stmsboot`" on page 62

System Fails to Boot During `stmsboot`

During the reboot following `stmsboot` enable (`-e`), disable (`-d`), or update (`-u`) operations if the system fails to boot but lets you enter single user mode, perform the following to recover.

1. **Type the root password and enter single user mode.**

2. **Restart the mpxio-upgrade service.**

   ```
   # svcadm restart svc:/platform/sun4u/mpxio-upgrade:default
   ```

If it doesn't work follow the instructions in the next section to recover your original configuration.

System Crashes During `stmsboot`

During the reboot following `stmsboot` enable (`-e`), disable (`-d`), or update (`-u`) operations, if your system hangs, panics, or fails to boot follow the instructions below to recover your original configuration.

1. **Boot the system from another disk, CD, DVD, or over the network.**

2. **Run the fsck <your-root-device> command where <your-root-device> is your original root device.**

   ```
   # fsck <your-root-device>
   ```

3. **Mount your original root device.**

   ```
   # mount <your-root-device> /mnt
   ```

4. **Restore your original** `fp.conf` **file by performing one of the following steps:**

   If you ran `stmsboot -e` or `-d`:

   ```
   # cp /mnt/etc/mpxio/fp.conf /mnt/kernel/drv/fp.conf

   The /mnt/etc/mpxio/fp.conf file is a copy of your original fp.conf
   file that stmsboot command saved prior to updating your fp.conf
   file.
   ```

   If you ran the `stmsboot -u` command, and you modified the `fp.conf` file prior to running the `stmsboot` command, undo the changes you made to this file by editing `/mnt/kernel/drv/fp.conf` file.

5. **Undo any other changes you made to the multipathing configuration prior to running the** `stmsboot` **command.**

   For example, if you modified the `scsi_vhci.conf` file, undo the changes you made to this file by editing the `/mnt/kernel/drv/scsi_vhci.conf` file.

   If you modified the device settings of your storage arrays, restore their original settings.

6. **Restore your original** /etc/vfstab **file:**

```
# cp /mnt/etc/mpxio/vfstab /mnt/etc/vfstab
```

The /mnt/etc/mpxio/vfstab file is a copy your original /etc/vfstab file that the stmsboot command saved prior to updating your vfstab file. A /mnt/etc/mpxio/vfstab file will not exist if the stmsboot command has not modified your vfstab file.

7. **Disable the mpxio-upgrade service:**

```
# /usr/sbin/svccfg -f /mnt/etc/mpxio/svccfg_recover
```

8. **Unmount the device.**

```
# umount /mnt
```

9. **Reboot.**

Index

M

modinfo, 39

P

physical device, 42
Promiscuous mode, 29

R

REPORT LUNS SCSI, 20

S

show_SCSI_LUN, 42, 49
single PCI HBA, 15
snoop(1M), 29
ssd, 39
st driver, 39
storage device, 19

T

To Detect Fabric Devices Visible on a Host, 48