

**EMULEX<sup>®</sup>**

*We network storage*

# **Boot Code**

*User Manual*

*For Emulex<sup>®</sup> HBAs*

Copyright© 2006 Emulex Corporation. All rights reserved worldwide. No part of this document may be reproduced by any means nor translated to any electronic medium without the written consent of Emulex Corporation.

Information furnished by Emulex Corporation is believed to be accurate and reliable. However, no responsibility is assumed by Emulex Corporation for its use; or for any infringements of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Emulex Corporation.

Emulex and LightPulse are registered trademarks, and AutoPilot Installer, AutoPilot Manager, BlockGuard, FibreSpy, HBAnyware, InSpeed, EZPilot, MultiPulse, SLI and SBOD are trademarks, of Emulex Corporation. All other brand or product names referenced herein are trademarks or registered trademarks of their respective companies or organizations.

Emulex provides this manual "as is" without any warranty of any kind, either expressed or implied, including but not limited to the implied warranties of merchantability or fitness for a particular purpose. Emulex Corporation may make improvements and changes to the product described in this manual at any time and without any notice. Emulex Corporation assumes no responsibility for its use, nor for any infringements of patents or other rights of third parties that may result. Periodic changes are made to information contained herein; although these changes will be incorporated into new editions of this manual, Emulex Corporation disclaims any undertaking to give notice of such changes.

Introduction.....	1
Using This Manual.....	1
Updating Boot Code .....	3
Using the HBAware Utility to Update Boot Code .....	3
Using Batch Mode to Update Boot Code.....	5
Using the HBAware Command-Line Interface to Update Boot Code .....	6
Using lputilnt to Update Boot Code .....	7
Using elxcfg to Update Boot Code.....	8
Using lputil to Update Boot Code .....	10
Using the EFI Utility to Update Boot Code.....	10
Using emlxadm to Update Boot Code .....	12
Using the emlxadm Command-Line Interface to Update Boot Code .....	14
Using the Offline Utility to Update Boot Code .....	15
Using lp6dutil to Update Boot Code.....	15
Using the lp6dutil Command-Line Interface to Update Boot Code .....	17
Enabling Boot Code.....	18
Using the HBAware Utility to Enable Boot Code.....	18
Using lputilnt to Enable Boot Code.....	20
Using lputil to Enable Boot Code .....	21
Using the Offline Utility to Enable Boot Code .....	21
Using lp6dutil to Enable Boot Code.....	21
Using the lp6dutil Command-Line Interface to Enable Boot Code.....	22
x86 BootBIOS.....	24
Using the BIOS Utility.....	24
Using the BIOS Utility to Enable an HBA to Boot from SAN and to Configure Adapter Parameters .....	24
Configure Boot Devices .....	32
Using Multi-Boot BIOS .....	34
Creating SAN Boot Disks .....	34
Installing Windows Server 2003 on a Fibre Channel Disk .....	34
Installing Linux on a Fibre Channel Disk .....	35
Installing Solaris (SFS Driver) on a Fibre Channel Disk .....	36
OpenBoot .....	46
Configuring OpenBoot.....	46
Configuring OpenBoot for Linux.....	46
Configuring OpenBoot for the Solaris lpfc Driver.....	46
Configuring OpenBoot for the Solaris SPARC SFS Driver .....	46
Configuring OpenBoot for AIX.....	46
Viewing OpenBoot Commands .....	47
Fibre Channel Boot Disks.....	47
Boot from SAN with Linux and Emulex HBAs.....	47
Creating a Fibre Channel Remote Boot Disk (Solaris lpfc).....	47
Making a Bootable System Disk (Solaris lpfc Driver) .....	50
Creating a Fibre Channel Remote Boot Disk (Solaris SPARC SFS Driver).....	52
EFIBoot .....	57
Configuring EFIBoot.....	57
Accessing the EFI Utility .....	57
Conventions for the EFI Utility.....	59
Making Configuration Changes Effective .....	59
Configuration Tasks.....	59
Configuring a Boot Device .....	64

Installing the Server.....	67
Install Windows Server 2003 (Itanium 64-bit) on a Fibre Channel Disk.....	67
Booting from SAN with Linux and Emulex HBAs.....	68
EFIBoot Reference.....	68
Driver Handle Table Column Definitions.....	68
Topology Information.....	68
Valid AL_PA Values.....	69
Installing the EFI Utility.....	69
Unloading EFIBoot.....	70
Troubleshooting.....	74
Troubleshooting x86 BootBIOS.....	74
The Bootup Message Does Not Appear As the System Boots.....	74
Retry This Adapter Message.....	74
Cannot Mount Root File System Message (Solaris SFS Driver with MPxIO Enabled).....	74
Cannot Find UNIX Kernel Message (Solaris SFS Driver).....	74
No Such Partition Message (Solaris SFS Driver).....	74
Troubleshooting OpenBoot.....	75
The System Cannot mount or fsck /etc/vfstab a Fibre Channel Boot Disk (Solaris lpfc Driver).....	75
A Loaded File Is Not Executable (Solaris lpfc Driver).....	75
The System Hangs or Reboots After Displaying Driver Information (Solaris lpfc Driver).....	75
Fibre Channel Disk Is Not Found (Solaris lpfc Driver).....	75
Troubleshooting EFIBoot.....	75
Diagnostics Overview.....	75
Showing Available Drivers and Controllers.....	76
Running Diagnostics.....	78

## Introduction

This document discusses the capabilities of booting servers directly from disk operating system images located on a storage area network (SAN) via Emulex® LightPulse® host bus adapters (HBAs), by using Emulex Fibre Channel boot code. It describes the Emulex-boot-from-SAN implementation and its operation with distinct hardware and operating system requirements.

Booting from SAN allows servers on a storage network to boot their operating system directly from a SAN storage device, typically identified by its World Wide Port Name (WWPN) and a logical unit number (LUN) located on the device. By extending the server system boot basic input/output system (BIOS), boot from SAN functionality is provided by the boot BIOS contained on an Emulex HBA in the server. When properly configured, the HBA then permanently directs the server to boot from a LUN on the SAN as if it were a local disk.

Emulex provides the following types of boot code:

- x86 BootBIOS works with the existing BIOS on x64 and x86 systems.
- OpenBoot works with the existing system BIOS on Solaris SPARC systems using the `lpfc` driver; Linux PowerPC systems; and AIX systems. OpenBoot is also called *FCode*.
- EFIBoot for Intel Itanium 64-bit systems. EFIBoot provides 64-bit system boot capability through the use of the EFI (Extensible Firmware Interface) Shell.

In addition, Emulex is unique in providing *Universal Boot* and *Pair Boot* code images that contain multiple types of boot code. These images provide multi-platform support for boot from SAN, offering enhanced server management and reduced system costs. Universal Boot and Pair Boot transparently determine your system platform type and automatically execute the proper boot code image in the HBA. These code images reside in HBA flash memory, allowing easier HBA portability and configuration between servers.

Universal Boot contains the following types of boot code:

- x86 BootBIOS
- OpenBoot
- EFIBoot

Pair Boot contains the following types of boot code:

- x86 BootBIOS
- EFIBoot

## Using This Manual

---

This document provides instructions for installing and using all of the types of boot code provided by Emulex. The manual sections that you will refer to depend on the type of boot code you are using:

- For x64 and x86 systems (x86 BootBIOS):
  1. See *Updating Boot Code* on page 3 to update boot code to a newer version.
  2. See *Enabling Boot Code* on page 18 to enable the boot code.
  3. See *Using the BIOS Utility* on page 24 to use the BIOS utility to configure boot options and to enable the HBA to boot from a SAN.
  4. See *Configure Boot Devices* on page 32 to configure boot devices.
  5. See *Installing Windows Server 2003 on a Fibre Channel Disk* on page 34 or *Installing Linux on a Fibre Channel Disk* on page 35 to create a Fibre Channel boot disk.

- For SPARC and PowerPC systems (OpenBoot):
  1. See *Updating Boot Code* on page 3 to update boot code to a newer version.
  2. See *Enabling Boot Code* on page 18 to enable the boot code.
  3. See *Configuring OpenBoot* on page 46 to configure OpenBoot.
  4. See *Fibre Channel Boot Disks* on page 47 to create a Fibre Channel boot disk.
- For Intel Itanium 64-bit systems (EFIBoot):
  1. See *Updating Boot Code* on page 3 to update boot code to a newer version.
  2. See *Enabling the HBA to Boot from SAN* on page 59 to enable the HBA to boot from a SAN.
  3. See *Configuration Tasks* on page 59 to configure boot options.
  4. See *Installing the Server* on page 67 to create a Fibre Channel boot disk.

## Updating Boot Code

A variety of Emulex utilities allow you to update boot code to a newer version (or to install it). The utility that you will use depends on the operating system and, in some cases, driver.

Operating System	HBAnyware utility	lputilnt <sup>1</sup>	elxcfg <sup>2</sup>	lputil	EFI utility <sup>3</sup>	emlxadm	Offline utility <sup>4</sup>	lp6dutil <sup>5</sup>
Windows	X	X	X				X	X
Linux	X			X			X	X
Solaris lpfc				X			X	X
Solaris SFS						X		
AIX <sup>6</sup>								

<sup>1</sup> SCSIport Miniport and Storport Miniport drivers only

<sup>2</sup> FC Port driver only

<sup>3</sup> Intel Itanium 64-bit systems only

<sup>4</sup> Requires an x64 or x86 system and Microsoft MS-DOS or FreeDOS (will not run in a DOS or command window)

<sup>5</sup> Requires an x64 or x86 system and Microsoft MS-DOS (will not run in a DOS or command window)

<sup>6</sup> The boot code on AIX systems is referred to as Open Firmware. To install Open Firmware on AIX PCI bus systems, you must use the IBM driver. See the IBM documentation, or visit the IBM Web site for more information.

After you decide which utility to use, see the appropriate procedure:

- HBAnyware™ utility: See *Using the HBAnyware Utility to Update Boot Code* on page 3.
- lputilnt: See *Using lputilnt to Update Boot Code* on page 7.
- elxcfg: See *Using elxcfg to Update Boot Code* on page 8.
- lputil: See *Using lputil to Update Boot Code* on page 10.
- EFI utility: See *Using the EFI Utility to Update Boot Code* on page 10.
- emlxadm: See *Using emlxadm to Update Boot Code* on page 12.
- Offline utility: See *Using the Offline Utility to Update Boot Code* on page 15.
- lp6dutil: See *Using lp6dutil to Update Boot Code* on page 15.

## Using the HBAnyware Utility to Update Boot Code

Prerequisites:

- One of the following drivers is installed properly:
  - SCSIport Miniport
  - Storport Miniport
  - FC Port
  - Emulex driver for Linux
  - Emulex lpfc driver for Solaris
- The HBAnyware utility has been installed properly.
- The boot code image file has been downloaded from the Emulex Web site and extracted to a folder on a local drive.
- If the HBA is already connected to a boot device, the system is in a state in which this type of maintenance can be performed:
  - I/O activity on the bus has been quieted.

- Cluster software, or any other software that relies on the HBA to be available, has been stopped or paused.

To update boot code using the HBAnyware utility:

1. Start the HBAnyware utility:
  - In Windows:
    - Click Start, Programs and HBAnyware.
    - From the command line, type `HBAnyware` and press the <Enter> key.
  - In Linux or Solaris:
    - `su` to root and run the following script:  
`/usr/sbin/hbanyware/hbanyware`
2. In the discovery-tree (left pane), click the HBA on which you want to update the boot code. The General tab (Figure 1) appears.

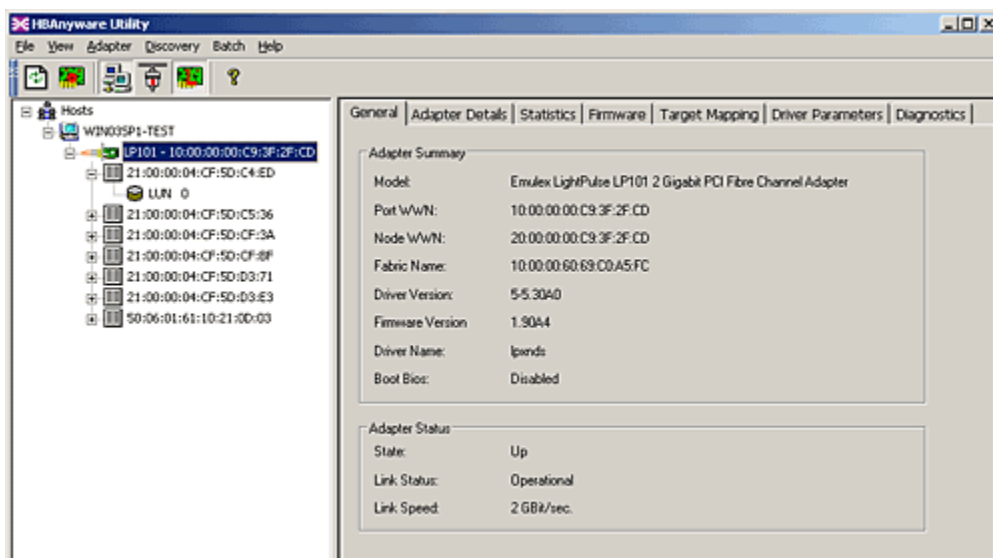


Figure 1: General Tab (Example)

3. In the property tabs (right pane), select the Firmware tab (Figure 2).

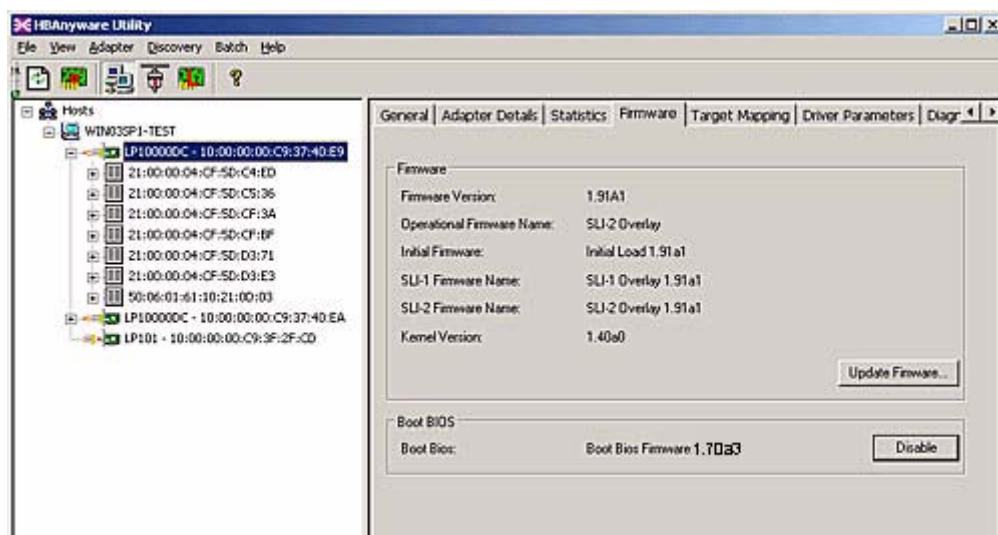


Figure 2: Firmware Tab (Example)



- For HBAAnyware versions earlier than 3.0, skip to step 5. In version 3.0 or later, click **Update Firmware**. The message in Figure 3 appears. Click **Yes**, and skip to step 6.

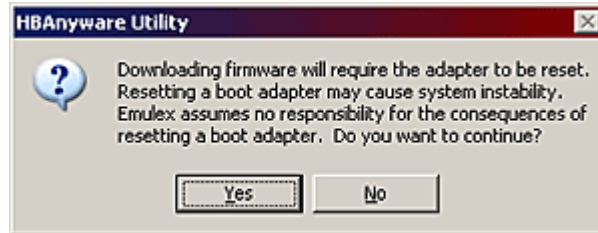


Figure 3: Reset Message

- In versions earlier than 3.0, click **Browse**.
- In HBAAnyware version 3.0 or later, the Firmware Download dialog box appears (Figure 4). In versions earlier than 3.0, the Select Firmware File dialog box is displayed.

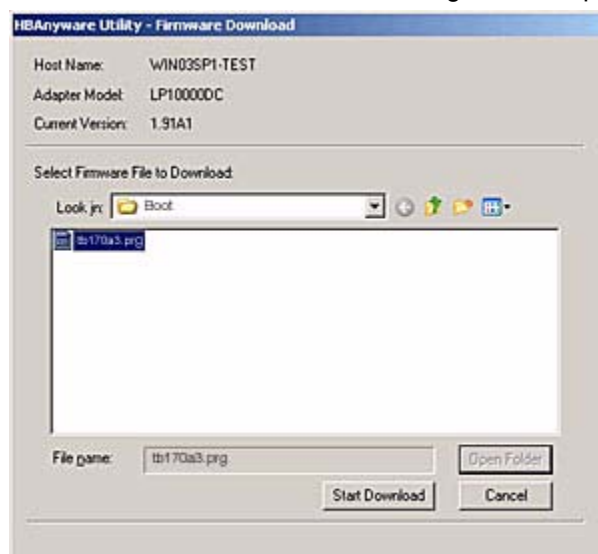


Figure 4: Firmware Download Dialog Box (Example)

- Navigate to the extracted boot code image file you want to download. Boot code images have a .PRG extension. Select the file and click **Start Download** (version 3.0 or later) or **OK** (versions earlier than 3.0). A status bar shows the progress of the download and indicates when the download is complete. In versions earlier than 3.0, the HBA in the discovery tree is displayed in red text, indicating that it is offline. It is displayed in black text when the update is complete.
- Click **Close**.
- Repeat steps 2 through 8 to update boot code on additional HBAs.

## Using Batch Mode to Update Boot Code

Batch mode allows you to update boot code on multiple HBAs in a single step. Batch mode is restricted to a single boot code image.

---

**Note:** No other HBA functions can be performed while boot code update in batch mode is in progress.

---

To update boot code using batch mode:

- Start the HBAAnyware utility.
- From the menu bar, select Batch and click Download Firmware.

3. Navigate to the extracted boot code image file you want to download. Boot code images have a .PRG extension. Select the boot code file and click **Open**. A tree-view shows all HBAs, and their corresponding hosts, that are compatible with the selected boot code file (Figure 5).

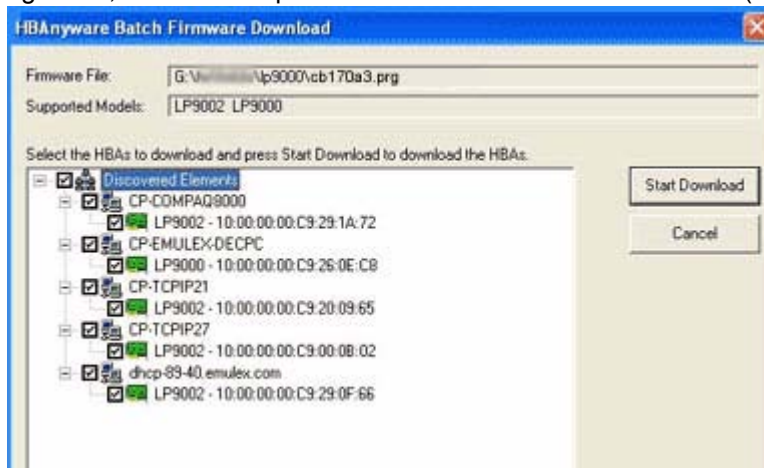


Figure 5: Selecting HBAs to Update (Example)

4. Click the box next to an HBA to select or remove the HBA from the batch process. Click the box next to a host to select or remove all eligible HBAs for the host from the batch process.
5. Click **Start Download**.
6. When downloading begins, the tree-view displays the progress. As the image for a selected HBA is being downloaded, it appears orange in the tree-view. When the download is complete, the entry changes to green (Figure 6). If the download failed, the entry changes to red.

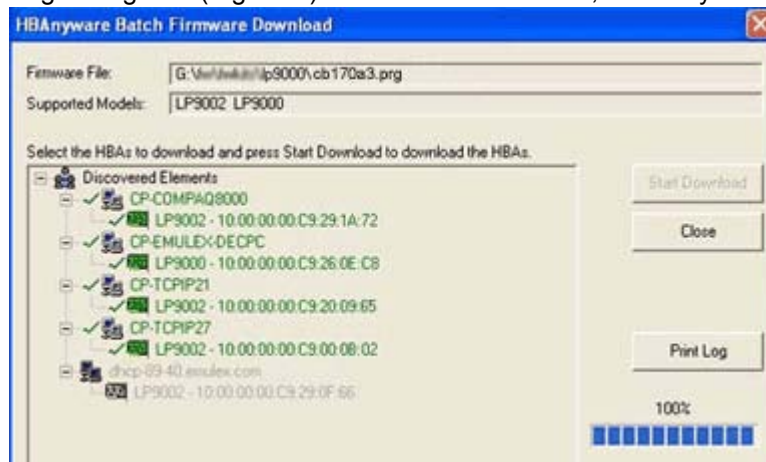


Figure 6: Download Complete (Example)

7. When downloading is finished, click **Print Log** to print a hard copy of the activity log.
8. Click **Close** to exit the batch procedure.

## Using the HBAware Command-Line Interface to Update Boot Code

To update boot code using the HBAware command-line interface:

1. Type the following at the command line, where *<wwpn>* is the worldwide port name (WWPN) of the HBA on which you want to update boot code and *<filename>* is the name of a boot code image (.PRG) file, such as BU502A0.PRG:

```
hbacmd download <wwpn> <filename>
```

For example, type:

```
hbacmd download 10:00:00:00:c9:2e:51:2e bu502a0.prg
```

Information similar to the following is displayed:

```
Downloading bu502a0.prg to hba 10:00:00:00:c9:2e:51:2e
```

2. Repeat step 1 to update boot code on additional HBAs.

## Using Iputilnt to Update Boot Code

Prerequisites:

- One of the following drivers is installed and loaded properly:
  - SCSIport Miniport
  - Storport Miniport
- Iputilnt is installed properly.
- The boot code image file has been downloaded from the Emulex Web site and extracted to a folder on a local drive.
- If the HBA is already connected to a boot device, the system is in a state in which this type of maintenance can be performed:
  - I/O activity on the bus has been quieted.
  - Cluster software, or any other software that relies on the HBA to be available, has been stopped or paused.

To update boot code using using Iputilnt:

1. Start Iputilnt: Click Start, Programs, Emulex and Iputilnt.
2. Select the desired HBA.
3. Select Firmware Maintenance from the Category list.

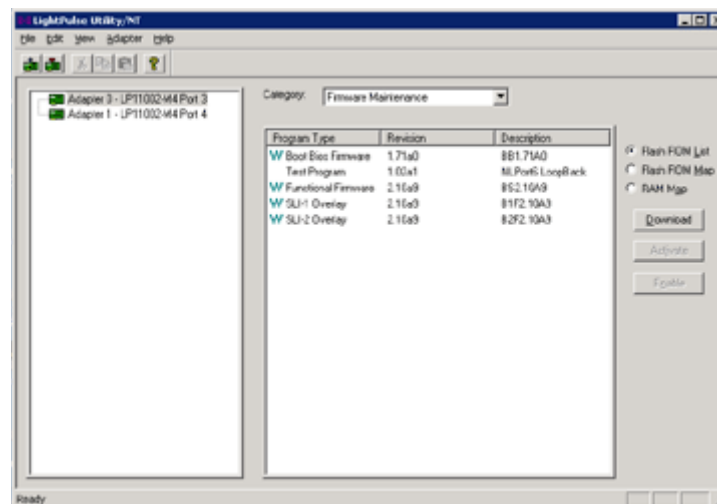


Figure 7: Firmware Maintenance

4. Click **Download**.

5. Navigate to the new boot code image file.

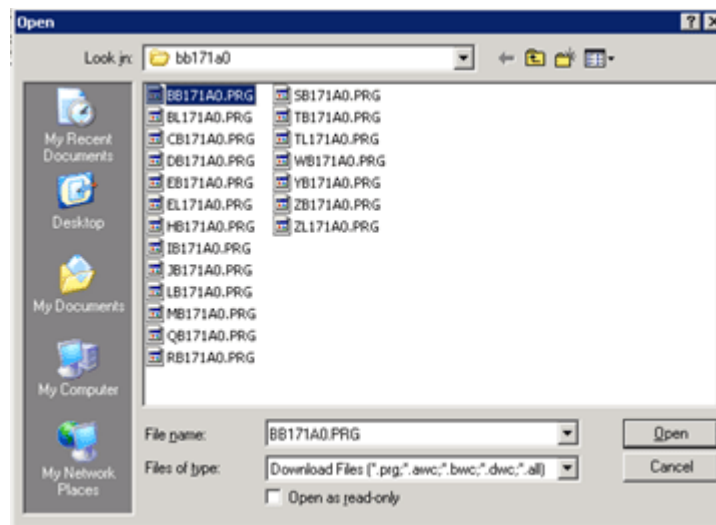


Figure 8: Download Boot Code Image

6. Click **Open**. The new boot code is transferred to flash read-only memory (ROM).
7. Repeat steps 2 through 6 to update boot code on additional HBAs.

## Using elxcfg to Update Boot Code

---

### Prerequisites:

- The FC Port driver is installed.
- elxcfg is installed properly.
- The boot code image file has been downloaded from the Emulex Web site and extracted to a folder on a local drive.
- If the HBA is already connected to a boot device, the system is in a state in which this type of maintenance can be performed:
  - I/O activity on the bus has been quieted.
  - Cluster software, or any other software that relies on the HBA to be available, has been stopped or paused.

To update boot code using using elxcfg:

1. Start elxcfg: Click Start, Programs, Emulex and elxcfg.

2. Locate and select an HBA from the Available Adapters list.

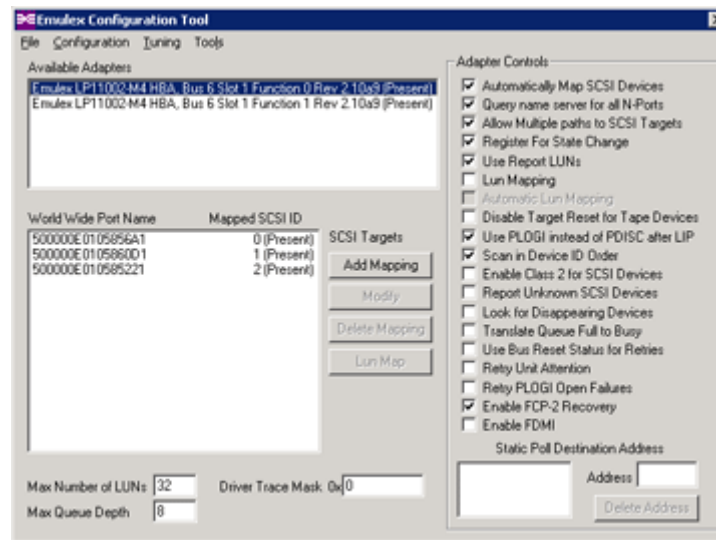


Figure 9: Selecting an Adapter

3. Select Download Firmware from the Tools menu. A browse window is displayed (Figure 10).

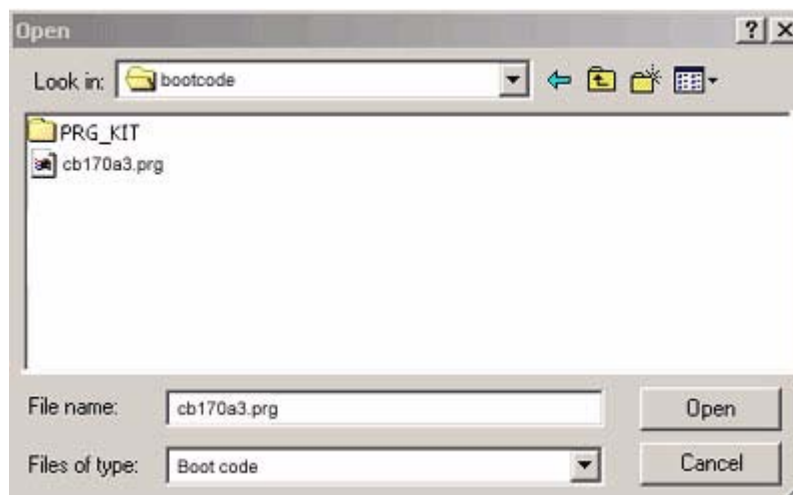


Figure 10: Browse Window

4. Navigate to the new boot code image file.
5. Click **Open**. The boot code is downloaded, and a confirmation window is displayed.



Figure 11: Confirmation Window

6. Click **OK** to close the confirmation window.
7. Repeat steps 2 through 6 to update the boot code on additional HBAs.

## Using lputil to Update Boot Code

---

### Prerequisites:

- The Emulex driver for Linux or Solaris is installed and loaded properly.
- lputil is installed properly.
- The boot code image file has been downloaded from the Emulex Web site and extracted to a folder on a local drive.
- If the HBA is already connected to a boot device, I/O activity on the bus has been quieted.

### To update boot code using lputil:

1. Start lputil by entering the complete path. The path in the example reflects the default installation path. If the installation path was modified, adjust the command appropriately.

```
/usr/sbin/lpfc/lputil
```

2. From the Main menu, enter **3**, Firmware Maintenance.
3. From the list that appears, choose the HBA onto which you want to update boot code.
4. Enter **1**, Load Firmware Image.
5. Enter the full path to the boot code image file. Boot code images have a .PRG extension. The new boot code is transferred to flash ROM.
6. Repeat steps 3 through 5 to update the boot code on additional HBAs.

## Using the EFI Utility to Update Boot Code

---

**Note:** If you are updating boot code on an HBA that is already attached to a boot device, it is recommended that you use the EFI utility, and not another utility such as the HBAnyware utility, to perform the update.

---

### Prerequisites:

- One of the following drivers is installed and loaded properly:
  - SCSIport Miniport
  - Storport Miniport
  - Emulex driver for Linux
- The EFI utility is installed (see *Installing the EFI Utility* on page 69 for more information).
- The boot code image file has been downloaded from the Emulex Web site and extracted to a folder on a local drive.

### To update EFIBoot using the EFI utility:

1. To view Emulex driver handle information, at the shell prompt type:

```
fs0:> drivers
```

A list of drivers is displayed (Figure 12).

```

D      T D
R      Y C I
V  VERSION E G G ND NC DRIVER NAME          IMAGE NAME
-----
10 00000010 B - - 4 10 PCI Bus Driver          PciBus
14 00000029 D - - 2 - Usb Ohci Driver          UsbOhci
15 00000010 B - - 2 2 USB Bus Driver          UsbBus
16 00001010 D X - 1 - Usb Keyboard Driver        UsbKb
17 00000010 ? - - - - Usb Mouse Driver          UsbMouse
18 00000010 D - - 1 - Usb Bot Mass Storage Driver      UsbBot
19 00000010 ? - - - - Usb Cbi0 Mass Storage Driver    UsbCbi0
1A 00000010 ? - - - - <UNKNOWN>                UsbCbi1
1B 00000010 D - - 1 - Generic USB Mass Storage Driver    UsbMassStorage
1C 00000010 ? - - - - UGA Console Driver          GraphicsConsole
1D 00000000 D - - 1 - PCI VGA Mini Port Driver          PciVgaMiniPort
1E 00000010 D - - 1 - VGA Class Driver                VgaClassDriver
1F 00000010 B - - 1 1 Serial 16550 UART Driver        Serial16550
20 00000010 B - - 1 1 Serial Terminal Driver          Terminal
21 00000010 D - - 2 - Platform Console Management Driver  ConPlatform
22 00000010 D - - 2 - Platform Console Management Driver  ConPlatform
23 00000010 B - - 2 2 Console Splitter Driver        ConSplitter
86 00031114 D X X 1 - Emulex SCSI Pass Thru Driver      \velxcli311a4.efi

```

Figure 12: Driver Information Example

**Note:** To view driver handle information one page at a time, type:  
`fs0:> drivers -b`

- To execute the Emulex EFI configuration utility, type the following, where *<driver handle>* is the number identified in the driver information as the *Emulex SCSI Pass Thru Driver*. (In the example in Figure 12, *<driver handle>* is 86.)

`drvcfg -s <driver handle>`

- Press *<Enter>*. An HBA listing is displayed (Table 1).

**Table 1: HBA Information Example**

001	LP1000DC	PCI-X	100 MHZ	BUS#: 40	Dev#: 01	Func# 00
002	LP8000	PCI	66 MHZ	BUS#: 61	Dev#: 02	Func# 00
003	LP8000	PCI	66 MHZ	BUS#: 61	Dev#: 02	Func# 00
004	LP952	PCI-X	100 MHZ	BUS#: A0	Dev#: 01	Func# 00
005	LP850	PCI	66 MHZ	BUS#: C0	Dev#: 02	Func# 00
006	LP982	PCI-X	100 MHZ	BUS#: C0	Dev#: 01	Func# 00
007	LP9802	PCI-X	100 MHZ	BUS#: E1	Dev#: 01	Func# 00
008	LP9802	PCI-X	100 MHZ	BUS#: E1	Dev#: 01	Func# 00

4. Locate the HBA on which you are installing EFIBoot. Select (highlight) the row and press <Enter>. The main EFI utility menu is displayed (Figure 13).

```

001: LP10000DC PCI-X 66MHz Bus#: 80 Dev#: 03 Func#: 01
    HBA Status: Ready      Boot Bios : Disabled
    EXP ROM   : Disabled   Firmware : TS1.91A1
  
```

---

1. Setup Utility
2. Firmware Update

<1/4> to Change Options. <Enter> to Select Option. <ESC> to Prev Menu.

Figure 13: EFI Utility Main Menu

5. Select **2**, Firmware Update, and press <Enter>. Enter the directory path for the boot code image (.PRG) file and press <Enter>. Enter the filename field and press <Enter> (Figure 14).

```

001: LP10000DC PCI-X 66MHz Bus#: 80 Dev#: 03 Func#: 01
    EXP ROM   : Disabled   SLI - 1  : T1D1.90A4
    Firmware  : TS1.90A4   SLI - 2  : T2D1.90A4
    Port Name : 00000000C93A5B7F Kernel   : 1.30A5
  
```

---

```

Directory Path : fs0:\temp\ebc
Filename       : te311a4.prg_
  
```

Figure 14: Directory Path and Filename Update Example

Functions are displayed as each system function is finished (Figure 15).

```

001: LP10000DC PCI-X 66MHz Bus#: 80 Dev#: 03 Func#: 01
    EXP ROM   : Disabled   SLI - 1  : T1D1.90A4
    Firmware  : TS1.90A4   SLI - 2  : T2D1.90A4
    Port Name : 00000000C93A5B7F Kernel   : 1.30A5
  
```

---

```

Directory Path : fs0:\temp\ebc
Filename       : te311a4.prg_
File Open      : Done
Validate Adapter : Done
Flash Erase    : Done
Flash Program  : Done
Wakeup Program : Done
  
```

Figure 15: Example of File Update Functions

6. Repeat steps 4 and 5 to update the boot code on additional HBAs.
7. Reboot the system.

## Using emlxadm to Update Boot Code

---

Prerequisites:

- The SFS driver for Solaris is installed and loaded properly.
- The emlxadm utility is installed.



- The boot code image file has been downloaded from the Emulex Web site and extracted to a folder on a local drive.
- If the HBA is already connected to a boot device, I/O activity on the bus has been quieted.

To update boot code using emlxadm:

1. Start the emlxadm utility by typing the name of the program without any command line arguments:

```
# /opt/EMLXemlxu/bin/emlxadm
```

2. After it is started, the emlxadm program scans the host system and prepares a list of qualified HBA ports to choose from. Qualified HBA ports are devices that attach to the SUN StorEdge SFS through the FP driver. After the list is prepared, the utility displays the following:

```
EMLXADM Device Management Utility, Version 1.00r
COPYRIGHT © 2004-2006 Emulex. All rights reserved.
```

Available HBA's:

1. /devices/pci@1e,600000/SUNW,qlc@3/fp@0,0:devctl (CONNECTED)
2. /devices/pci@1e,600000/SUNW,qlc@3,1/fp@0,0:devctl (NOT CONNECTED)
3. /devices/pci@1e,600000/SUNW,emlxs@2/fp@0,0:devctl (CONNECTED)
4. /devices/pci@1e,600000/SUNW,emlxs@2,1/fp@0,0:devctl (NOT CONNECTED)

Enter an HBA number or zero to exit:

3. You must choose from one of the available HBAs in the list by entering the appropriate number. In this example, if you enter 3, the utility displays the HBA device name selected and presents a list of command options:

```
HBA: /devices/pci@1e,600000/SUNW,emlxs@2/fp@0,0:devctl
```

Available commands:

[rev 2]

```
get_num_devs - Returns the number of FC devices seen by this HBA.
get_dev_list - Returns a list of FC devices seen by this HBA.
get_logi_params <wwpn> - Returns the login paramters for a specified FC device.
get_host_params - Return the host parameters.
get_sym_pname - Returns the symbolic port name of a device.
set_sym_pname <string> - Sets the symbolic port name for a device.
get_sym_nname - Returns the symbolic node name of a device.
set_sym_nname <string> - Sets the symbolic node name for a device.
dev_login <wwpn> - Performs an FC login to a device.
dev_logout <wwpn> - Performs an FC logout to a device.
get_state <wwpn> - Returns current Leadville state of a specified device.
dev_remove <wwpn> - Remove the FC device from Leadville management.
link_status <d_id> - Request link error status from a specified D_ID.
get_fcode_rev - Returns the current Fcode revision of the HBA.
download_fcode [filename] - Download the HBA fcode.
get_fw_rev - Returns the current firmware revision of the HBA.
download_fw [filename] - Download the HBA firmware.
get_boot_rev *Returns the current boot revision of the HBA.
download_boot [filename] *Download the HBA boot image.
get_dump_size - Returns the HBA's firmware core dump size.
force_dump - Force a firmware core dump on this HBA.
get_dump <-t,-b> <file> - Saves firmware core dump to a file.
get_topology - Returns the current FC network topology.
reset_link <wwpn,0> - Resets the link of a specified FC device.
reset_hard - Reset the HBA.
reset_hard_core - Reset the HBA firmware core.
diag <test> - Perform a diagnostic test on the HBA.
```

```

ns - Performs a complete query of the fabric name server.
parm_get_num - *Returns the total number of configurable parameters.
parm_get_list - *Returns a list of configurable parameters.
parm_get <label> - *Gets the value of a specified parameter in the driver.
parm_set <label> <val> - *Sets the value of a specified parameter in the driver.
msgbuf all or <number> [-i interval] - *Returns the driver's internal message
log.
get_host_attrs - Returns the host adapter and port attributes.
get_port_attrs <index>, <wwn> or all - Returns the port attributes.
get_path <index> - Returns the adapter path.
get_vpd - *Returns the adapter's Vital Product Data (VPD).
q - Exits this program.
h - Returns this help screen.
hba - Select another hba.
p - Repeat previous command.

```

\*Emulex adapters only

```
emlxadm>
```

4. Type the following, where *<filename>* is the name of a boot code image (.PRG) file, such as BU502A0.PRG:

```
download_boot <filename>
```

Information similar to the following is displayed:

```
Image Components: REL type      size=143416
                  DWC file:      BOOT: version=03845054, 5.02a0
```

```
Current: Boot: 5.01a2
New:      Boot: 5.02a0 143416 (0x23038) bytes
```

Are you sure you want to download this image? (y or n):

5. Press **y**. The following is displayed:

```
Downloading...
```

```
Done.
```

---

**Note:** If the file name is not provided, the program attempts to identify the adapter model, then downloads a default boot image file if one is available.

---

6. Repeat steps 3 through 5 to update the boot code on additional HBAs.
7. Press **q** to exit the emlxadm utility.

## Using the emlxadm Command-Line Interface to Update Boot Code

To update boot code using the emlxadm command-line interface:

1. Type the following at the command line, where *<device>* is the the full device name of the HBA on which you want to update boot code and *<filename>* is the name of a boot code image (.PRG) file, such as BU502A0.PRG:

```
emlxadm <device> <filename>
```

For example, type (all on one line):

```
emlxadm /devices/pci@1e,600000/SUNW,emlxs@2/fp@0,0:devctl download_boot
BU502A0.PRG
```

Information similar to the following is displayed:

```
Image Components: REL type      size=143416
                  DWC file:      BOOT: version=03845054, 5.02a0
```

```
Current: Boot: 5.01a2
New:     Boot: 5.02a0 143416 (0x23038) bytes
```

Are you sure you want to download this image? (y or n):

2. Press **y**. The following is displayed:

```
Downloading...
```

```
Done.
```

---

**Note:** If the file name is not provided, the program attempts to identify the adapter model, then downloads a default boot image file if one is available.

---

3. Repeat steps 1 and 2 to update the boot code on additional HBAs.

## Using the Offline Utility to Update Boot Code

---

Prerequisites:

- The Offline utility application file (DOSLPCFG.EXE) is available on the local drive.
- The boot code image file has been downloaded from the Emulex Web site and extracted to a folder on the local drive.

To update boot code using the Offline utility:

1. Boot the system with MS-DOS or FreeDOS.

---

**Note:** You cannot start the Offline utility at the DOS or command prompt in Windows.

---

2. From the directory where the DOSLPCFG.EXE file resides:
  - a. To download a firmware image file to all HBAs of the same specified name, type the following, where *<adapter name>* is the name of the HBAs on which you want to update boot code and *<filename>* is the name of a boot code image (.PRG) file, such as BU502A0.PRG:

```
doslpcfg download a=<adapter name> i=<filename>
```

The following example downloads the BU502A0.PRG boot code image file to all LP11000 HBAs:

```
doslpcfg download a=lp11000 i=c:\image\bu502a0.prg
```

- b. To download a firmware image file to an HBA specified by its number, type the following, where *<adapter number>* is the adapter number of the HBA on which you want to update boot code and *<filename>* is the name of a boot code image (.PRG) file, such as BU502A0.PRG:

```
doslpcfg download n=<adapter number> i=<filename>
```

The following example downloads the BU502A0.PRG boot code image file to HBA # 6:

```
doslpcfg download n=6 i=c:\image\bu502a0.prg
```

3. Repeat step 2 to update the boot code on additional HBAs.

## Using Ip6dutil to Update Boot Code

---

Prerequisites:

- The Ip6dutil.exe file is available on the local drive.
- The boot code image file has been downloaded from the Emulex Web site and extracted to a folder on the local drive.

To update boot code using lp6dutil:

1. Boot the system with DOS.

**Note:** You cannot start lp6dutil at the DOS or command prompt in Windows.

2. From the directory where the lp6dutil.exe file resides, enter the following command:

```
lp6dutil
```

An Introduction screen, which has a menu bar near the top and an lp> prompt near the bottom, is displayed.

3. Press and hold down <Alt> and press L. The Flash Menu is displayed (Figure 16).



Figure 16: Flash Menu

4. Press D. The Download Image window is displayed (Figure 17).

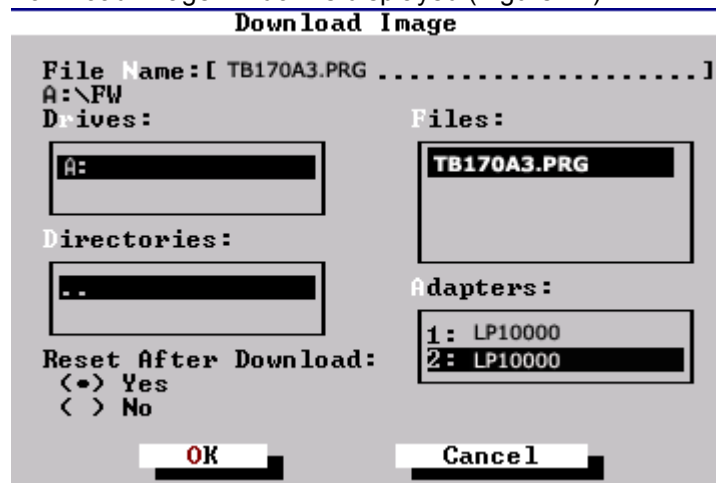


Figure 17: Download Image Window

5. Highlight the drive on which the boot code image is located, and press <Enter>.
6. Highlight the directory in which the boot code image is located, and press <Enter>.
7. Highlight the boot code image filename in the Files list, and press <Enter>. Boot code images have a .PRG extension.
8. Highlight the HBA on which you want to update boot code in the Adapters list, and press <Enter>.
9. The adapter resets automatically after the image is downloaded. Set Reset After Download to No if you are loading several individual files to one adapter.
10. Hold down the <Alt> key and press O to select OK. The download process begins. Various steps of the download process are displayed along with the results of the download (success or error).
11. Repeat steps 3 through 10 to update the boot code on additional HBAs.

## Using the lp6dutil Command-Line Interface to Update Boot Code

To update boot code using the lp6dutil command-line interface:

1. Boot the system with DOS.
2. From the directory where the lp6dutil.exe file resides, enter the following command:  

```
lp6dutil
```

An Introduction screen, which has a menu bar near the top and an lp> prompt near the bottom, is displayed.
3. At the lp> prompt, do one of the following:
  - a. To download a firmware image file to all HBAs of the same specified name, type the following, where *<adapter name>* is the name of the HBAs on which you want to update boot code and *<filename>* is the name of a boot code image (.PRG) file, such as BU502A0.PRG:  

```
download a=<adapter name> i=<filename>
```

The following example downloads the BU502A0.PRG boot code image file to all LP11000 HBAs:  

```
download a=lp11000 i=c:\image\bu502a0.prg
```
  - b. To download a firmware image file to an HBA specified by its number, type the following, where *<adapter number>* is the adapter number of the HBA on which you want to update boot code and *<filename>* is the name of a boot code image (.PRG) file, such as BU502A0.PRG:  

```
download n=<adapter number> i=<filename>
```

The following example downloads the BU502A0.PRG boot code image file to HBA # 6:  

```
download n=6 i=c:\image\bu502a0.prg
```
4. Repeat step 3 to update the boot code on additional HBAs.

## Enabling Boot Code

Before you can use the BIOS utility to configure boot code on an x64 or x86 system, the boot code must be enabled.

A variety of Emulex utilities allow you to enable the boot code. The utility that you will use depends on the operating system and, in some cases, driver or system hardware.

Operating System	HBAnyware utility	lputilnt <sup>1</sup>	lputil	Offline utility <sup>2</sup>	lp6dutil <sup>3</sup>
Windows <sup>4</sup>	X	X		X	X
Linux	X		X	X	X
Solaris lpfc	X		X	X	X
Solaris SFS <sup>5</sup>					

<sup>1</sup> SCSIport Miniport and Storport Miniport drivers only

<sup>2</sup> Requires an x64 or x86 system and Microsoft MS-DOS or FreeDOS (will not run in a DOS or command window)

<sup>3</sup> Requires an x64 or x86 system and Microsoft MS-DOS (will not run in a DOS or command window)

<sup>4</sup> For the FC Port driver, you must use the HBAnyware utility, the Offline utility or lp6dutil to enable boot code; elxcfg does not provide this functionality.

<sup>5</sup> Boot code for Solaris SFS systems is enabled automatically when it is installed, so no utility is needed

After you decide which utility to use, see the appropriate procedure:

- HBAnyware utility: See *Using the HBAnyware Utility to Enable Boot Code* on page 18.
- lputilnt: See *Using lputilnt to Enable Boot Code* on page 20.
- lputil: See *Using lputil to Enable Boot Code* on page 21.
- Offline utility: See *Using the Offline Utility to Enable Boot Code* on page 21.
- lp6dutil: See *Using lp6dutil to Enable Boot Code* on page 21.

## Using the HBAnyware Utility to Enable Boot Code

---

Prerequisites:

- One of the following drivers is installed properly:
  - SCSIport Miniport
  - Storport Miniport
  - FC Port
  - Emulex driver for Linux
- The HBAnyware utility has been installed properly

To enable boot code using the HBAnyware utility:

1. Start the HBAnyware utility:
  - In Windows:
    - Click Start, Programs and HBAnyware.
    - From the command line, type `HBAnyware` and press the <Enter> key.

- In Linux:
  - su to root and run the following script:  
/usr/sbin/hbanyware/hbanyware
- 2. In the discovery-tree (left pane), click the HBA on which you want to enable the boot code. The General tab (Figure 18) appears.

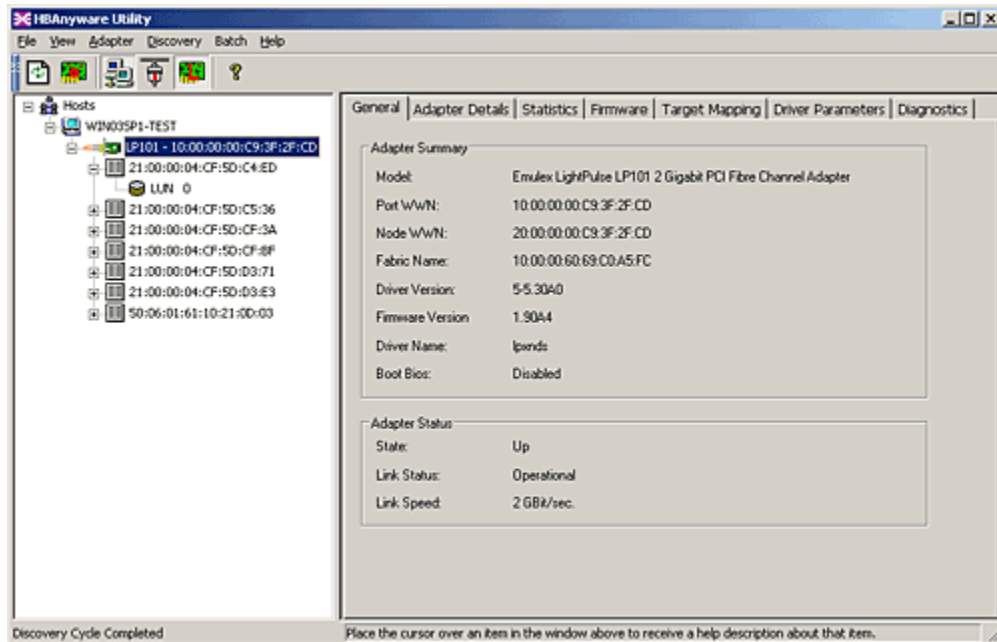


Figure 18: General Tab (Example)

3. In the property tabs (right pane), select the Firmware tab (Figure 19).

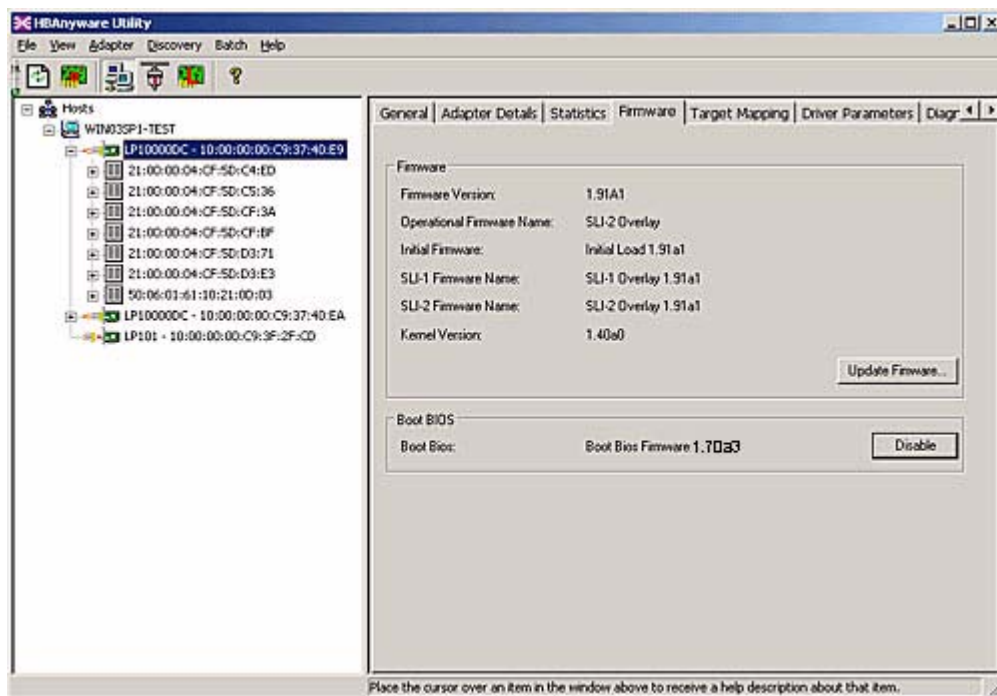


Figure 19: Firmware Tab (Example)

- If the **Disable** button appears, the boot code is already enabled; you do not need to enable it. Otherwise, click **Enable**; the **Enable** button changes to **Disable**

---

**Note:** If the state of the boot code has changed, this change is reflected immediately on the General tab (version 3.0 or later) or Details tab (versions earlier than 3.0).

---

## Using Iputilnt to Enable Boot Code

---

Prerequisites:

- One of the following drivers is installed and loaded properly:
  - SCSIport Miniport
  - Storport Miniport
- Iputilnt is installed properly.

To enable boot code using using Iputilnt:

- Start Iputilnt: Click Start, Programs, Emulex and Iputilnt.
- Select the desired HBA.
- Select Firmware Maintenance from the Category list.

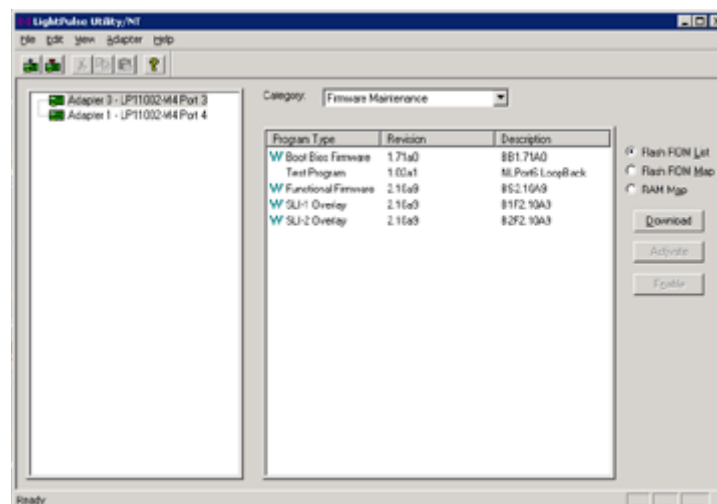


Figure 20: Firmware Maintenance

- Click Boot Bios Firmware (Figure 21).

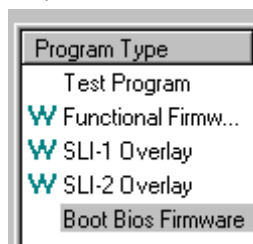


Figure 21: Program Type List

- If the **Disable** button appears, the boot code is already enabled; you do not need to enable it. Otherwise, click **Enable**; the **Enable** button changes to **Disable** and the letter W appears to the left of "Boot Bios Firmware" in the Program Type list. This indicates that the boot code is enabled.



## Using lputil to Enable Boot Code

---

Prerequisites:

- The Emulex driver for Linux is installed and loaded properly.
- lputil is installed properly.

To enable boot code using lputil:

1. Start lputil by entering the complete path. The path in the example reflects the default installation path. If the installation path was modified, adjust the command appropriately.

```
/usr/sbin/lpfc/lputil
```

2. From the Main menu, press **3**, Firmware Maintenance.
3. Press **6**, Boot BIOS Maintenance.
4. Press **1**, Enable Boot BIOS, to enable the boot code. (If item 1 is Disable Boot BIOS, the boot code is already enabled; you do not need to enable it.)

## Using the Offline Utility to Enable Boot Code

---

Prerequisites:

- The Offline utility application file (DOSLPCFG.EXE) is available on the local drive.

To enable boot code using the Offline utility:

1. Boot the system with MS-DOS or FreeDOS.

---

**Note:** You cannot start the Offline utility at the DOS or command prompt in Windows.

---

2. From the directory where the DOSLPCFG.EXE file resides, type the following, where *<adapter number>* is the adapter number of the HBA on which you want to enable boot code:

```
doslpcfg listboot n=<adapter number>
```

A display similar to the following appears. The number following boot is the index number, which you will use in step 3. Each version of boot code that is installed on the HBA will have a unique index number.

```
boot 1 (enabled): RB1.60A7  
boot 2 (disabled): RB1.52A1
```

3. From the directory where the DOSLPCFG.EXE file resides, type the following, where *<adapter number>* is the adapter number of the HBAs on which you want to enable boot code and *<index>* is the index number for the version of boot code that you identified in step 2.

```
doslpcfg enableboot n=<adapter number> i=<index>
```

The following example enables the boot code for index number 1 on HBA #6:

```
doslpcfg enableboot n=6 i=1
```

## Using lp6dutil to Enable Boot Code

---

Prerequisites:

- The lp6dutil.exe file is available on the local drive.

To enable or disable boot code using lp6dutil:

1. Boot the system with DOS.

---

**Note:** You cannot start lp6dutil at the DOS or command prompt in Windows.

---

- From the directory where the lp6dutil.exe file resides, enter the following command:

```
lp6dutil
```

An Introduction screen, which has a menu bar near the top and an lp> prompt near the bottom, is displayed.

- Press and hold down <Alt> and press L. The Flash Menu is displayed (Figure 22).



Figure 22: Flash Menu

- Press B. The Change Boot BIOS State window is displayed (Figure 23).



Figure 23: Change Boot BIOS State Window

- Highlight the HBA in the Adapters list.
- Highlight the boot code image file in the BIOS Images list.
- Tab to **Change** and press <Enter>. If boot code was disabled, it is now enabled for the selected HBA. If it was enabled, it is now disabled.
- Hold down the <Alt> key and press O to select **OK** to close the window.

## Using the lp6dutil Command-Line Interface to Enable Boot Code

To enable boot code using the lp6dutil command-line interface:

- Boot the system with DOS.
- From the directory where the lp6dutil.exe file resides, enter the following command:

```
lp6dutil
```

An Introduction screen, which has a menu bar near the top and an lp> prompt near the bottom, is displayed.

- At the lp> prompt, type the following, where <adapter number> is the adapter number of the HBA on which you want to enable boot code:

```
listboot n=<adapter number>
```

A display similar to the following appears. The number following *boot* is the index number, which you will use in step 3. Each version of boot code that is installed on the HBA will have a unique index number.

```
boot 1 (enabled): RB1.60A7
boot 2 (disabled): RB1.52A1
```

4. At the `lp>` prompt, type the following, where *<adapter number>* is the adapter number of the HBAs on which you want to enable boot code and *<index>* is the index number for the version of boot code that you identified in step 3.

```
enableboot n=<adapter number> i=<index>
```

The following example enables the boot code for index number 1 on HBA #6:

```
enableboot n=6 i=1
```

## x86 BootBIOS

This chapter provides procedures for:

- Using the BIOS utility (see *Using the BIOS Utility* on page 24)
- Creating SAN boot disks (see *Creating SAN Boot Disks* on page 34)

Troubleshooting information is provided in *Troubleshooting x86 BootBIOS* on page 74.

Follow the procedures in this section to set up remote boot on all x64 and x86 systems.

### Using the BIOS Utility

---

The BIOS utility allows you to do the following:

- Enable an HBA to boot from SAN and change HBA parameters (such as topology, PLOGI retry timer and auto scan setting); see *Using the BIOS Utility to Enable an HBA to Boot from SAN and to Configure Adapter Parameters* on page 24
- Configure boot devices; see *Configure Boot Devices* on page 32

---

**Note:** This section reflects the most recent release of the BIOS utility. Some selections may not be available if you are using an older version of the utility.

---

### Using the BIOS Utility to Enable an HBA to Boot from SAN and to Configure Adapter Parameters

- Default settings are acceptable for most installations.
- In the BIOS utility, press **d** to reset the selected adapter to default values.
- The default topology is auto topology with loop first. For FC-AL, each HBA has a default AL\_PA of 01 (Hex). Change this topology setting, if necessary, before you configure boot devices.
- All HBAs or boot drives can be configured to other AL\_PAs than their default values.
- If more than one HBA is in the system with the same PCI bus number, and each has a boot drive attached, the first PCI-scan HBA will be the boot HBA. The first HBA is usually in the lowest PCI slot in the system. This information can be viewed from the BIOS utility.

Prerequisites:

- The boot code is loaded on the HBA as described in *Updating Boot Code* on page 3.
- The boot code is enabled as described in *Enabling Boot Code* on page 18.

To use the BIOS utility:

1. Turn on the computer.
2. Press and hold down <Alt> or <Ctrl> and press **E** immediately (within five seconds) when the bootup message to start the BIOS utility is displayed. An HBA listing is displayed (Figure 24).

---

**Note:** If the bootup message does not appear, you must enable x86 BootBIOS. See *Enabling Boot Code* on page 18 for more information.

---

```

Emulex Light Pulse BIOS Utility, ZB1.71a0
Copyright (c) 1997-2006 Emulex. All rights reserved.

Emulex Adapters in the System:

1. LP11000:          PCI Bus:15 Device:06 Function:00
2. LP11000:          PCI Bus:15 Device:06 Function:01
3. LPe11000-M4:     PCI Bus:01 Device:00 Function:00
  
```

Figure 24: HBA Listing

3. Select the adapter to configure by entering the appropriate number. The main configuration menu is displayed (Figure 25).

```

Adapter 01:          PCI Bus:15 Device:06 Function:00

LP11000:            I/O Base: 6C00 Firmware Version: BS2.10A9
Port Name: 10000000 C927744E Node Name: 20000000 C927744E
Topology: Auto Topology: Loop First (Default)
The BIOS of this Adapter is Disable

1. Configure Boot Devices
2. Configure This Adapter's Parameters
  
```

Figure 25: Main Configuration Menu

4. Select **2** to configure the adapter's parameters. The adapter configuration menu is displayed (Figure 26).

```

Adapter 01:          PCI Bus:15 Device:06 Function:00

LP11000:            I/O Base: 6C00 Firmware Version: BS2.10A9
Port Name: 10000000 C927744E Node Name: 20000000 C927744E
Topology: Auto Topology: Loop First (Default)
The BIOS of this Adapter is Disable

1. Enable or Disable BIOS
2. Change Default ALPA Of This Adapter
3. Change PLOGI Retry Timer (+Advanced Option+)
4. Topology Selection (+Advanced Option+)
5. Enable or Disable Spinup Delay (+Advanced Option+)
6. Auto Scan Setting (+Advanced Option+)
7. Enable or Disable EDD 3.0 (+Advanced Option+)
8. Enable or Disable Start Unit Command (+Advanced Option+)
9. Enable or Disable Environment Variable (+Advanced Option+)
A. Enable or Disable Auto Boot Sector (+Advanced Option+)
B. Link Speed Selection (+Advanced Option+)
  
```

Figure 26: Adapter Configuration Menu

## Adapter Configuration Menu Summaries

1. **Enable or Disable BIOS.** Enable the selected HBA to boot from SAN. See *Using the BIOS Utility* on page 24 for more information. You must enable at least one HBA to boot from SAN to use remote boot functionality.
2. **Change Default ALPA of this adapter.** This option applies only to arbitrated loop (FC-AL). The factory default is 01 (hexadecimal). See *Change the Default AL\_PA* on page 26 for more information.
3. **Change PLOGI Retry Timer (+Advanced Option+).** Set the interval for the port log in retry timer. The factory default is No PLOGI Retry: 0 msec. See *Change the PLOGI Retry Timer* on page 27 for more information.
4. **Topology Selection (+Advanced Option+).** If it is necessary to change the topology, do so before you configure boot devices. See *Change Topology* on page 27 for more information.

5. **Enable or Disable Spinup Delay (+Advanced Option+)**. Enable or disable the spinup delay. The factory default is disabled. See *Enable or Disable the Spinup Delay* on page 28 for more information.
  6. **Auto Scan Setting (+Advanced Option+)**. Enable the first device in the boot entry list to issue a Name Server Inquiry. The factory default is disabled. See *Set Auto Scan* on page 29 for more information.
  7. **Enable or Disable EDD 3.0 (+Advanced Option+)**. Enable or disable the Enhanced Disk Driver (EDD) option, which is available on Intel IA-64 servers only. The factory default is disabled. See *Enable/Disable EDD 3.0* on page 29 for more information.
  8. **Enable or Disable Start Unit Command (+Advanced Option+)**. You must know the specific LUN to issue the SCSI Start Unit Command. The factory default is disabled. See *Enable or Disable the Start Unit Command* on page 30 for more information.
  9. **Enable or Disable Environment Variable (+Advanced Option+)**. Enable or disable the ability to set the boot controller order, if the system supports the environment variable. The factory default is disabled. See *Enable or Disable the Environment Variable* on page 30 for more information.
- A. **Auto Boot Sector (+Advanced Option+)**. (Applies only to HP MSA1000 arrays.) This option automatically defines the boot sector of the target disk for the migration boot process. The factory default is disabled. See *Enable or Disable Auto Boot Sector* on page 31 for more information.
- B. **Link Speed Selection**. Select the HBA link speed. The factory default is Auto Select. See *Link Speed Selection* on page 31 for more information.

## Change the Default AL\_PA

This option allows you to change the AL\_PA (Arbitrated Loop Physical Address) of the selected adapter. The default value of the AL\_PA for the adapter BIOS is 00 (hex).

---

**Note:** This option applies only to arbitrated loop (FC-AL).

---

To change the default AL\_PA:

1. Start the BIOS utility and select the adapter.
2. When the main menu opens select **2**, Configure This Adapter's Parameters. The adapter configuration menu is displayed (Figure 26).
3. Select **2**, Change Default ALPA of this adapter (Figure 27).

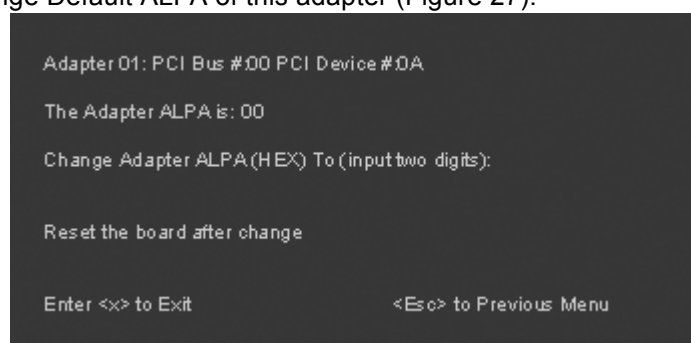


Figure 27: Change Default ALPA Screen

4. Type the AL\_PA, in hexadecimal notation, to which you want to change the default.

5. Press **x** to exit the BIOS utility and reboot the system.

---

**Note:** If the HBA's AL\_PA is changed, it will not show on the NVRAM AL\_PA until the system has been reset.

---

## Change the PLOGI Retry Timer

This option allows you to set the interval for the PLOGI (port log in) retry timer. This option is especially useful for Tachyon-based RAID arrays. Under very rare occasions, a Tachyon-based RAID array will reset itself and the port will go offline temporarily in the loop. When the port comes to life, the PLOGI retry interval scans the loop to discover this device. This default setting is No PLOGI Retry: 0 msec.

To set the interval for the PLOGI retry timer:

1. Start the BIOS utility and select the adapter.
2. When the main menu opens select **2**, Configure This Adapter's Parameters. The adapter configuration menu is displayed (Figure 26).
3. Select **3**, Change PLOGI Retry Timer (+Advanced Option+) (Figure 28).

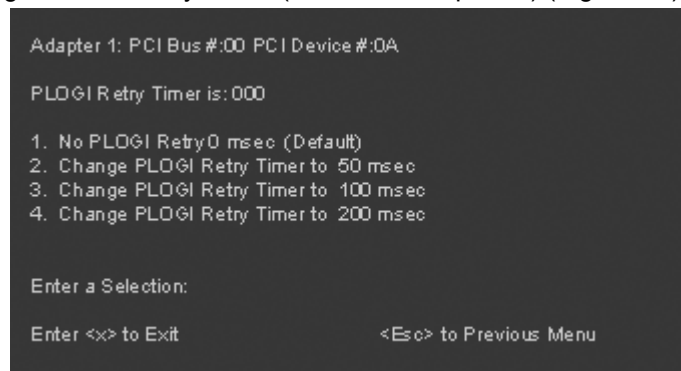


Figure 28: Change the PLOGI Retry Timer Screen

4. Select the PLOGI retry timer interval. The time it takes for one PLOGI to scan the whole loop (if 126 AL\_PA are on the loop) is shown below:
  - 50 msec will take 5 to 6 seconds per device.
  - 100 msec will take 12 seconds per device.
  - 200 msec will take 22 seconds per device.
5. Press **x** to exit the BIOS utility and reboot the system.

## Change Topology

This option allows you to select the topology for the HBA.

---

**Note:** The default topology is auto topology with loop first. Change this topology setting, if necessary, before configuring boot devices.

---

To select the HBA topology:

1. Start the BIOS utility and select the adapter.
2. When the main menu opens select **2**, Configure This Adapter's Parameters. The adapter configuration menu is displayed (Figure 26).

3. Select **4**, Topology Selection (+Advanced Option+) (Figure 29).

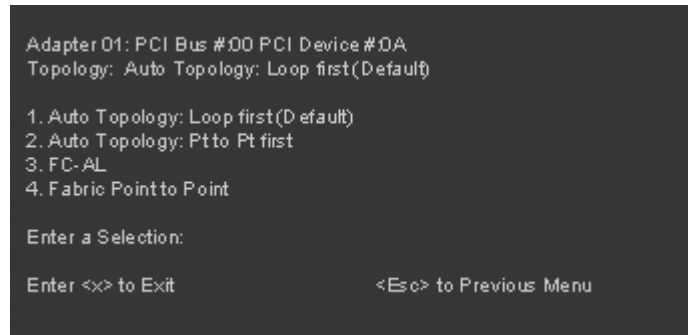


Figure 29: Topology Menu

4. Select the topology for the HBA:
  - Select **1** for auto topology with loop first.
  - Select **2** for auto topology with point-to-point first.
  - Select **3** for FC-AL.
  - Select **4** for fabric point-to-point.
5. Press **x** to exit the BIOS utility and reboot the system.

### Enable or Disable the Spinup Delay

This option allows you to enable or disable the spinup delay. The factory default setting is disabled.

If at least one boot device has been defined, and the spinup delay is enabled, the BIOS searches for the first available boot device.

- If a boot device is present, the BIOS boots from it immediately.
- If a boot device is not ready, the BIOS waits for the spinup delay and, for up to three additional minutes, continues the boot scanning algorithm to find another multi-boot device.

If no boot devices have been defined, and auto scan is enabled, then the BIOS waits for five minutes before scanning for devices.

- In a private loop, it attempts to boot from the lowest target AL\_PA it finds.
- In an attached fabric, it attempts to boot from the first target found in the NameServer data.

To enable or disable the spinup delay:

1. Start the BIOS utility and select the adapter.
2. When the main menu opens select **2**, Configure This Adapter's Parameters. The adapter configuration menu is displayed (Figure 26).
3. Select **5**, Enable or Disable Spinup Delay (+Advanced Option+) (Figure 30).

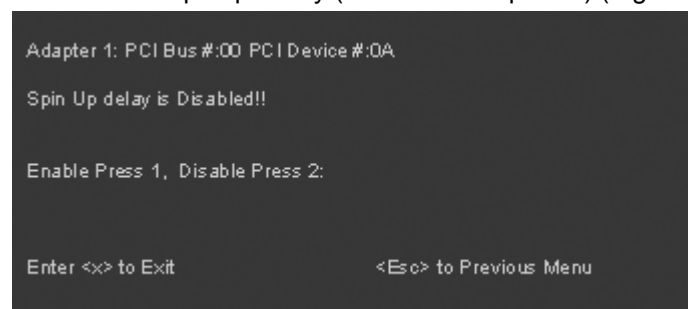


Figure 30: Enable or Disable Spinup Delay Screen



4. Select **1** to enable the spinup delay, or select **2** to disable it.
5. Press **x** to exit the BIOS utility and reboot the system.

### Set Auto Scan

This option allows you to set auto scan. The factory default setting is disabled. Auto Scan is available only if none of the eight boot entries is configured to boot via DID or WWPN. It is strongly recommended that you use the Configure Boot Devices menu to configure eight boot entries for fabric point-to-point, public loop or private loop configurations.

With auto scan enabled, the first device will issue a Name Server Inquiry. The boot device will be either the first DID, LUN 0, or not LUN 0 device returned, depending on the option you select. Only this device will be the boot device and it will be the only device exported to the Multi-boot menu.

To set auto scan:

1. Start the BIOS utility and select the adapter for which you want to set auto scan.
2. When the main menu opens select **2**, Configure This Adapter's Parameters. The adapter configuration menu is displayed (Figure 26).
3. Select **6**, Auto Scan Setting (+Advanced Option+) (Figure 31).

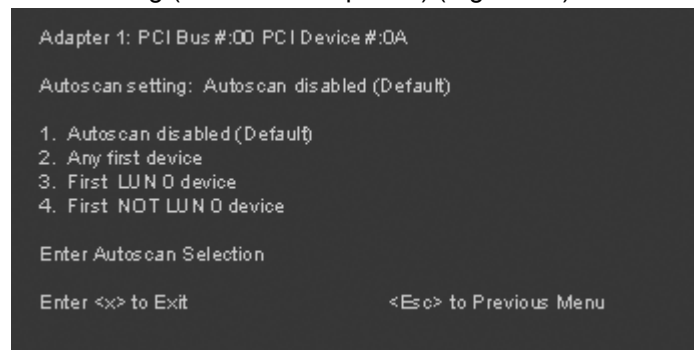


Figure 31: Set Auto Scan Screen

4. Select the auto scan option:
  - Select **1** to disable auto scan.
  - Select **2** to scan for any first device. The first adapter issues a name server inquiry, and the first D\_ID from the inquiry becomes the boot device. The adapter attempts to log in to a public loop first. If it fails, it logs in to a private loop. The first successfully scanned device becomes the boot device, and only this device is exported to the multi-boot menu.
  - Select **3** to scan for the first LUN 0 device.
  - Select **4** to scan for the first device other than LUN 0.
5. Press **x** to exit the BIOS utility and reboot the system.

### Enable/Disable EDD 3.0

This option allows you to enable or disable the Enhanced Disk Drive (EDD) option, available on Intel Itanium servers only. Enabling EDD 3.0 displays the path to the boot device.

The default setting is disabled.

---

**Note:** An x86 series system could hang during Windows 2000 Server installation if EDD 3.0 is enabled.

---

To enable or disable EDD 3.0:

1. Start the BIOS utility and select the adapter for which you want to enable or disable EDD 3.0.
2. When the main menu opens select **2**, Configure This Adapter's Parameters. The adapter configuration menu is displayed (Figure 26).
3. Select **7**, Enable or Disable EDD 3.0 (+Advanced Option+) (Figure 32).

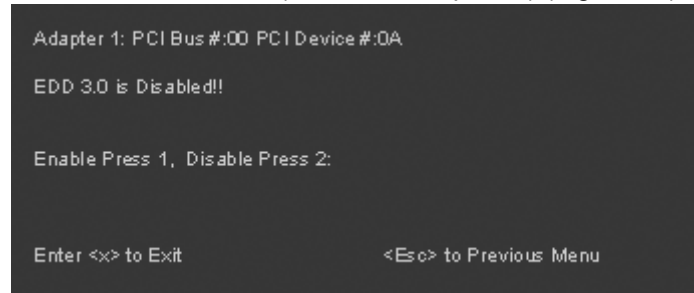


Figure 32: Enable or Disable EDD 3.0 Screen

4. Select **1** to enable EDD 3.0, or press **2** to disable it.
5. Press **x** to exit the BIOS utility and reboot the system.

### Enable or Disable the Start Unit Command

This command is used to issue the SCSI start unit command. You must know the specific LUN to issue the SCSI start unit command. The default setting is disabled.

To enable or disable the start unit command:

1. Start the BIOS utility and select the adapter for which you want to enable or disable the start unit command.
2. When the main menu opens select **2**, Configure This Adapter's Parameters. The adapter configuration menu is displayed (Figure 26).
3. Select **8**, Enable or Disable Start Unit Command (+Advanced Option+) (Figure 33).

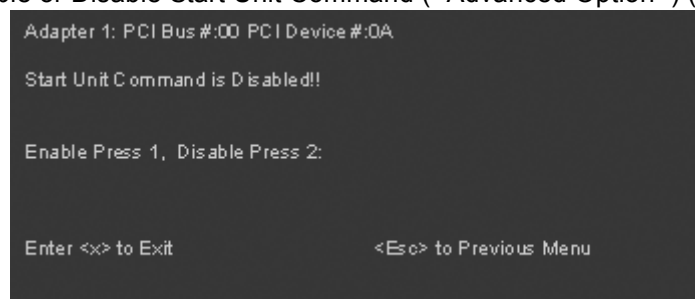


Figure 33: Enable or Disable Start Unit Command Screen

4. Select **1** to enable the start unit command, or **2** to disable it.
5. Press **x** to exit the BIOS utility and reboot the system.

### Enable or Disable the Environment Variable

This option allows you to set the boot controller order if the system supports the environment variable. The default setting is disabled.

To enable or disable the environment variable:

1. Start the BIOS utility and select the adapter for which you want to enable or disable the environment variable.

2. When the main menu opens select **2**, Configure This Adapter's Parameters. The adapter configuration menu is displayed (Figure 26).
3. Select **9**, Enable or Disable Environment Variable (+Advanced Option+) (Figure 34).

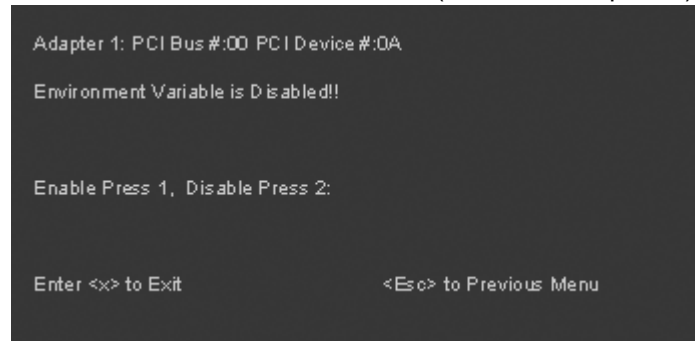


Figure 34: Enable or Disable Environment Variable Screen

4. Select **1** to enable the environment variable, or **2** to disable it.
5. Press **x** to exit the BIOS utility and reboot the system.

### Enable or Disable Auto Boot Sector

This option automatically defines the boot sector of the target disk for the migration boot process, which applies only to HP MSA1000 arrays. If there is no partition on the target, the default boot sector format is 63 sectors. The default setting is disabled.

To enable or disable auto sector format select:

1. Start the BIOS utility and select the adapter for which you want to enable or disable the command.
2. When the main menu opens select **2**, Configure This Adapter's Parameters. The adapter configuration menu is displayed (Figure 26).
3. Select **A**, Enable or Disable Auto Boot Sector (+Advanced Option+) (Figure 35).



Figure 35: Enable or Disable Auto Sector Format Select Screen

4. Select **1** to enable auto boot sector, or **2** to disable it.
5. Press **x** to exit the BIOS utility and reboot the system.

### Link Speed Selection

This option allows you to select the adapter's link speed. The default link speed is Auto Select, which automatically selects the link speed.

To select the adapter's link speed:

1. Start the BIOS utility and select the adapter for which you want to select the link speed.
2. When the main menu opens select **2**, Configure This Adapter's Parameters. The adapter configuration menu is displayed (Figure 26).

3. Select **B**, Link Speed Selection (Figure 36).

```

Adapter B1:          PCI Bus:15 Device:06 Function:00

Link Speed is: Auto Select (Default)

0. Auto Select (Default)
1. 1 Gigabaud
2. 2 Gigabaud
4. 4 Gigabaud

```

Figure 36: Link Speed Selection Screen

4. Select one of the following:
  - **0** for Auto Select (the adapter's speed is selected automatically based on its model).
  - **1** for 1 Gigabaud.
  - **2** for 2 Gigabaud.
  - **4** for 4 Gigabaud.
5. Press **x** to exit the BIOS utility and reboot the system.

## Configure Boot Devices

This option supports FC\_AL: (public loop and private loop) and fabric point-to-point. When operating in loop (FC\_AL) topology, the system automatically determines whether you are configured for a public or private loop. The BIOS looks for a fabric loop (FL\_Port) first. If a fabric loop is not detected, the BIOS looks for a private loop. For the "Configure Boot Devices" option, the eight boot entries will be zero by default (<d> key).

To configure boot devices:

1. Start the BIOS utility and select the adapter for which you want to configure boot devices.
2. When the Main menu is displayed, select **1**, Configure Boot Devices.

A list of 8 boot devices is shown (Figure 37). We recommend that only the bootable devices be configured. The primary boot device is the first entry shown, and it is the first bootable device.

- If the first boot entry fails due to a hardware error, the system can boot from the second bootable entry.

- If the second boot device fails, the system will boot from the third bootable entry and so on.

```

Adapter 03: S_ID: 000017 PCI Bus:01 Device:00 Function:00

List of Saved Boot Devices:

1. Used DID:000000 WWPN:21000011 C6810689 LUN:00 Primary Boot
2. Used DID:000000 WWPN:21000011 C6810629 LUN:00
3. Unused DID:000000 WWPN:00000000 00000000 LUN:00
4. Unused DID:000000 WWPN:00000000 00000000 LUN:00
5. Unused DID:000000 WWPN:00000000 00000000 LUN:00
6. Unused DID:000000 WWPN:00000000 00000000 LUN:00
7. Unused DID:000000 WWPN:00000000 00000000 LUN:00
8. Unused DID:000000 WWPN:00000000 00000000 LUN:00

Select a Boot Entry: _

Enter <x> to Exit <Esc> to Previous Menu

```

Figure 37: List of Saved Boot Devices

3. Select a boot entry. The screen in Figure 38 is displayed.

```

Adapter 03: S_ID: 000017 PCI Bus:01 Device:00 Function:00

00. Clear selected boot entry!!
01. ALPA:01(7C) WWPN:21000011 C6810689 LUN:00EACATE ST336854FC 4001
02. ALPA:02(7B) WWPN:21000011 C6810629 LUN:00EACATE ST336854FC 4001
03. ALPA:04(7A) WWPN:21000011 C6810684 LUN:00EACATE ST336854FC 4001
04. ALPA:08(79) WWPN:21000011 C68105F0 LUN:00EACATE ST336854FC 4001
05. ALPA:0F(78) WWPN:21000011 C68108C1 LUN:00EACATE ST336854FC 4001
06. ALPA:10(77) WWPN:21000011 C68108BA LUN:00EACATE ST336854FC 4001

```

Figure 38: Device Selection List

**Note:** To minimize the amount of time needed to locate the boot device, we recommend that you select the drive with the lowest AL\_PA as the boot device.

4. Select 00 to clear the selected boot entry, or select a device to configure booting via WWPN or DID. If you select a device, you will be asked for the starting LUN.
5. Enter the starting LUN (Figure 39). The starting LUN can be any number between 0 and 255.

```

Adapter 03: S_ID: 000017 PCI Bus:01 Device:00 Function:00

ALPA:04 SelectID:7A WWPN:21000011 C6810684

01. LUN:00 SEAGATE ST336854FC 4001

```

Figure 39: LUN Listing

A possible 256 LUNs can be defined per adapter, but the screen displays up to 16 consecutive LUNs at a time. In front of each entry, B#D or B#W specifies the boot entry number and whether the device boots via DID or WWPN. For example, B1D means that boot entry one will boot via DID. B2W means that boot entry two will boot via WWPN.

6. Type the two digits corresponding to the entry you are selecting. Another menu displays the following options:

1. Boot this device via WWPN
2. Boot this device via DID

7. Select the desired boot method.

If you select to boot the device via WWPN, the WWPN of the earlier selected entry is saved in the flash. However, during the initial BIOS scan, it will issue a Name Server Inquiry GID\_PN (Get Port Identifier). Then, based on this DID, it will continue with the remainder of the scanning.

If you select to boot this device via DID, the earlier selected entry will be saved in the flash ROM.

## Using Multi-Boot BIOS

Multi-boot BIOS is in compliance with BIOS Boot Specification (BBS). The system must have a Multi-boot system BIOS in order to take advantage of this feature. Multi-boot BIOS allows you to select any boot disk in the system BIOS setup menu. The boot disk can be a Fibre Channel drive, a SCSI drive, an IDE drive or a floppy drive. The Emulex BIOS supplies the first eight drives to the system BIOS menu. The Multi-boot BIOS can override the Fibre Channel drive that is selected in the BIOS utility.

For example, the system has only eight Fibre Channel disks. The boot disk has AL\_PA 02. However, you can select AL\_PA 23 in the system BIOS setup menu. The boot device will be the Fibre Channel disk with AL\_PA 23 instead of AL\_PA 02, as is set in the BIOS utility.

If your system supports Multi-boot BBS, the boot disk (drive C) is the first entry in Multi-boot on the system BIOS setup menu. The list of entries is determined by the list of configured boot entries in the BIOS utility. For example:

```
Adapter 1: boot_entry0, boot_entry1
Adapter 2: boot_entry2, boot_entry3
```

The order of boot entries that is exported to Multi-boot (BBS) is

```
boot_entry0, boot_entry1, boot_entry2, and boot_entry3.
```

However, Multi-boot allows changing the boot order in the server BIOS, which allows any disk to become the C drive.

## Creating SAN Boot Disks

---

This section explains how to create SAN boot disks in the following operating systems:

- Windows Server 2003 and Windows 2000 Server: See *Installing Windows Server 2003 on a Fibre Channel Disk* on page 34.
- Linux: See *Installing Linux on a Fibre Channel Disk* on page 35.
- Solaris SFS driver: See *Installing Solaris (SFS Driver) on a Fibre Channel Disk* on page 36.

## Installing Windows Server 2003 on a Fibre Channel Disk

This procedure installs a Windows operating system onto a previously unformatted Fibre Channel disk drive that is connected to the HBA. Although the examples in this section are specific to Windows Server 2003, the procedure for Windows 2000 Server is similar.

The computer's system BIOS may require that another controller take precedence over the Emulex HBA during boot. If this occurs, you must disconnect or disable the other adapter. This will allow you to configure and build your operating system on the drive connected to the Emulex HBA.

To install Windows 2003 Server on a Fibre Channel disk:

1. From <http://www.emulex.com>, download the distribution executable file for the latest version of the Emulex driver to your local drive. You can download the SCSIport Miniport driver, the Storport Miniport driver, or the FC Port driver. The file you download is an executable (.exe) file.
2. In Windows Explorer, double-click the distribution executable file.  
A window is displayed with driver version information.
3. Click **Next** to access the Location window, or click **Cancel** to close the window. If you click **Next**, the default installation location is displayed. Browse to a different location, if desired.
4. Click **Install** to continue the installation. A progress window is displayed. As each task is completed, the corresponding checkbox is automatically selected. After all tasks are completed, a confirmation window is displayed.
5. Clear the Start AutoPilot Installer checkbox, and click **Finish** to close the distribution executable file.
6. In Windows Explorer, navigate to the folder you specified in step 3.
7. In the \AutoPilot Installer\Drivers\*drivername* folder, open the folder that corresponds to your computer type, such as x86.  
*drivername* is the type of driver you downloaded (for example, Storport Miniport).
8. Copy the files that are in this folder onto a formatted floppy disk.
9. Boot the target system with the Windows Server 2003 Setup Media.
10. Follow the prompts that appear on the screen. The following message is displayed:  

```
Press F6 if you need to install a third party SCSI or RAID driver
```
11. Press <F6>. The following message is displayed:  

```
Setup could not determine the type of one or more mass storage
devices installed on your system, or you have chosen to
manually specify an adapter.
Currently, Setup will load support for the following mass
storage device(s):
<additional information>
S=Specify Additional Device    ENTER=continue    F3=Exit.
```
12. Press <S> to specify additional devices. The following message is displayed:  

```
Please insert the disk labeled Manufacturer-supplied hardware
support disk into Drive A:. Press ENTER when ready.
ENTER=continue    ESC=cancel    F3=Exit
```
13. Insert the floppy disk onto which you copied the driver files in step 8, and press <Enter>. A list of devices is displayed.
14. Select your HBA from the list and press <Enter>. The following message is displayed:  

```
Setup will load support for the following mass storage device(s):
Emulex xxxxxx PCIxx Fibre Channel Adapter
```
15. Press <Enter>.
16. Remove the floppy disk that contains the driver files.
17. Follow the remaining prompts to complete the installation.

## Installing Linux on a Fibre Channel Disk

Use the driver on the Linux distribution disk to boot the system. If necessary, you can then update the driver to the desired version.

## Installing Solaris (SFS Driver) on a Fibre Channel Disk

This section explains how to install Solaris (SFS driver) on a Fibre Channel disk:

- With Multiplexed I/O (MPxIO) enabled: See *Remote SFS SAN Boot Setup with MPxIO Enabled* on page 36.
- With MPxIO disabled: See *Remote SFS SAN Boot Setup with MPxIO Disabled* on page 41.

### Remote SFS SAN Boot Setup with MPxIO Enabled

This section provides the procedure to enable a disk to boot from a StorEdge SAN Foundation Software (SFS) SAN with MPxIO enabled.

Prerequisites:

- Solaris 10 (x64 and x86) and its latest patches are installed (see the operating system documentation).
- The Emulex SFS device driver for Solaris is installed (see the Sun documentation that accompanies the device driver).
- The Emulex FCA utilities kit is installed.
- x86 BootBIOS is loaded and enabled.
- A SAN disk with formatted capacity at least as large as the local disk is available.
- The `mpxio-disable` parameter is set to `no` in the `/kernel/drv/fp.conf` file (see the Sun MPxIO documentation).
- The correct storage device is inserted in the `scsi_vhci.conf` file.

To set up remote boot, do the following:

1. Start the format utility. At the prompt, type:

```
format
```

2. Select the local drive.

3. Type:

```
partition
```

4. Display the partition layout with the print command. Type:

```
print
```

Leave this window open.

5. Select the LUN you are going to use as the SAN boot disk (not the local drive) using the `luxadm probe` command. Record LUN information, which is used throughout this procedure. LUN 1 is used in the example.

```
# luxadm probe
Found Fibre Channel device(s):
  Node WWN:50060e8003823800  Device Type:Disk device
    Logical Path:/dev/rdisk/c3t50060E800382380000008238000000F0d0s2
  Node WWN:50060e8003823800  Device Type:Disk device
    Logical Path:/dev/rdisk/c3t50060E800382380000008238000000ECd0s2
```

6. Format the selected LUN with the format command. Specify the path or select from an available disk selection list.

In the example, the following starts the format utility and specifies the target disk:

```
# format /dev/rdisk/c3t50060E800382380000008238000000F0d0s2
```

Or, to select from a list, enter the format command:



```
# format
```

The following message and listing appears:

```
Searching for disks...done
```

```
AVAILABLE DISK SELECTIONS:
```

- ```
0. c1d0 <DEFAULT cyl 38729 alt 2 hd 64 sec 63>
   /pci@0,0/pci-ide@7/ide@0/cmdk@0,0
1. c3t50060E800382380000008238000000ECd0 <DEFAULT cyl 1193 alt 2 hd 255 sec 63>
   /scsi_vhci/disk@g50060e800382380000008238000000ec
2. c3t50060E800382380000008238000000F0d0 <DEFAULT cyl 1193 alt 2 hd 255 sec 63>
   /scsi_vhci/disk@g50060e800382380000008238000000f0
```

```
Specify disk (enter its number): 2
```

When prompted, select the disk you want to format. Use all of the disk space for the Solaris active partition. In the example, disk 2 has been selected and the following appears:

```
selecting c3t50060E800382380000008238000000F0d0
[disk formatted]
```

7. After you have formatted the LUN, the Format menu appears. Start the fdisk utility. Type:

```
format> fdisk
```

No fdisk table exists. Accept the default that creates:

```
a 100% "SOLARIS System" partition
```

8. After the Solaris system partition is created, results similar to the following appear:

```
Total disk size is 1196 cylinders
Cylinder size is 16065 (512 byte) blocks
Cylinders
Partition  Status  Type           Start  End  Length  %
=====  =====  =====
1         Active  Solaris2       1     1195  1195    100
```

```
SELECT ONE OF THE FOLLOWING:
```

- ```
1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)
```

```
Enter Selection:
```

When prompted, select option 5 to exit disk configuration. Partition 1 of the disk you are going to use as the SAN boot disk is now the active partition and you are still in the format utility.

9. The Format menu appears. While still in the format utility, prepare to modify the newly created partition to look like the source drive. Type:

```
partition> print
```

```
Current partition table (original):
```

```
Total disk cylinders available: 1193 + 2 (reserved cylinders)
```

Part	Tag	Flag	Cylinders	Size	Blocks
0	unassigned	wm	0	0	(0/0/0) 0
1	unassigned	wm	0	0	(0/0/0) 0
2	backup	wu	0 - 1192	9.14GB	(1193/0/0) 19165545
3	unassigned	wm	0	0	(0/0/0) 0
4	unassigned	wm	0	0	(0/0/0) 0
5	unassigned	wm	0	0	(0/0/0) 0
6	unassigned	wm	0	0	(0/0/0) 0

```

7 unassigned  wm      0      0      0      (0/0/0)      0
8 boot        wu      0 -    0      7.84MB      (1/0/0)      16065
9 unassigned  wm      0      0      0      (0/0/0)      0

```

10. Modify the partition of the SAN boot disk with the modify command to match the partition of the source drive that is displayed in the window you opened in step 4:

```
partition> modify
```

```
Select partitioning base:
```

```
0. Current partition table (original)
```

```
1. All Free Hog
```

```
Choose base (enter number) [0]? 1
```

11. At the prompt, enter 1 to start the All Free Hog utility. Note that the All Free Hog utility gives a printout of the current size of the disk. Results similar to the following appear:

Part	Tag	Flag	Cylinders	Size	Blocks
0	root	wm	0	0	(0/0/0) 0
1	swap	wu	0	0	(0/0/0) 0
2	backup	wu	0 - 1192	9.14GB	(1193/0/0) 19165545
3	unassigned	wm	0	0	(0/0/0) 0
4	unassigned	wm	0	0	(0/0/0) 0
5	unassigned	wm	0	0	(0/0/0) 0
6	usr	wm	0	0	(0/0/0) 0
7	unassigned	wm	0	0	(0/0/0) 0
8	boot	wu	0 - 0	7.84MB	(1/0/0) 16065
9	alternates	wm	0	0	(0/0/0) 0

```
Do you wish to continue creating a new partition table based on above table[yes]?
```

When prompted, type 'yes'. In the example, the following appears:

```

Enter size of partition '1' [0b, 0c, 0.00mb, 0.00gb]: 2gb
Enter size of partition '3' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '4' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '5' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '6' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '7' [0b, 0c, 0.00mb, 0.00gb]:

```

Part	Tag	Flag	Cylinders	Size	Blocks
0	root	wm	1 - 930	7.12GB	(930/0/0) 14940450
1	swap	wu	931 - 1192	2.01GB	(262/0/0) 4209030
2	backup	wu	0 - 1192	9.14GB	(1193/0/0) 19165545
3	unassigned	wm	0	0	(0/0/0) 0
4	unassigned	wm	0	0	(0/0/0) 0
5	unassigned	wm	0	0	(0/0/0) 0
6	usr	wm	0	0	(0/0/0) 0
7	unassigned	wm	0	0	(0/0/0) 0
8	boot	wu	0 - 0	7.84MB	(1/0/0) 16065
9	alternates	wm	0	0	(0/0/0) 0

```
Okay to make this the current partition table[yes]?
```

Accept the current partition table. You are then prompted to label the table. Enter the table name with quotation marks around it:

```
Enter table name (remember quotes): "disk"
```

When prompted, type 'y' to label the disk. Then enter the quit command. Type:

```

Ready to label disk, continue? y
partition> quit

```

12. Quit the format utility and create a new file system on the root partition that you created on the new SAN boot disk with the newfs command. In the example, results similar to the following appear:

```
# newfs /dev/rdisk/c3t50060E800382380000008238000000F0d0s0
newfs: /dev/rdisk/c3t50060E800382380000008238000000F0d0s0 last mounted as /ztmp
newfs: construct a new file system /dev/rdisk/
c3t50060E800382380000008238000000F0d0s0:
(y/n)? y
```

Type **y** to construct a new file system. In the example, the following appears:

```
Warning: 13230 sector(s) in last cylinder unallocated
/dev/rdisk/c3t50060E800382380000008238000000F0d0s0:      14940450 sectors in 989
cylinders of 240 tracks, 63 sectors
      7295.1MB in 142 cyl groups (7 c/g, 51.68MB/g, 6272 i/g)
super-block backups (for fsck -F ufs -o b=#) at:
   32, 105936, 211840, 317744, 423648, 529552, 635456, 741360, 847264, 953168,
 13979360, 14085264, 14191168, 14297072, 14402976, 14508880, 14614784,
 14720688, 14826592, 14932496,
```

13. Create additional file systems as needed.

14. Create a mount point on the local drive and mount the new file system of the boot device that you created in step 13:

```
# cd /
# mkdir rtmp
#
# mount /dev/dsk/c3t50060E800382380000008238000000F0d0s0 /rtmp
```

15. Use the ufsdump (backup) and ufsrestore (restore) commands to copy the contents of the root file system on the local drive to the new file system on the SAN boot disk. In the example, the following results appear:

```
# ufsdump 0f - /dev/rdisk/c1d0s0 | (cd /rtmp; ufsrestore rf -)
DUMP: Date of this level 0 dump: Thu Jan 12 10:01:13 2006
DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /dev/rdisk/c1d0s0 (s12100:/) to standard output.
DUMP: Mapping (Pass I) [regular files]
DUMP: Mapping (Pass II) [directories]
DUMP: Writing 32 Kilobyte records
DUMP: Estimated 6348042 blocks (3099.63MB).
DUMP: Dumping (Pass III) [directories]
DUMP: Dumping (Pass IV) [regular files]
Warning: ./lost+found: File exists
DUMP: 54.52% done, finished in 0:08
DUMP: 6348030 blocks (3099.62MB) on 1 volume at 3082 KB/sec
DUMP: DUMP IS DONE
```

16. Perform a backup and restore on any other file systems created during this process.

17. Change directories to the /ztmp/etc directory and modify the vfstab file. Replace the local disk instance with the new SAN boot disk instance. In the example, c1d0s0 is replaced with c3t50060E800382380000008238000000F0d0:

```
# pg vfstab
#device          device          mount          FS          fsck          mount          mount
#to mount        to fsck         point          type        pass         at boot options
#
fd               -              /dev/fd fd     -           no           -
/proc           -              /proc  proc    -           no           -
/dev/dsk/c3t50060E800382380000008238000000F0d0s1    -           -           swap        -
```

```
no -
/dev/dsk/c3t50060E800382380000008238000000F0d0s0
/dev/rdisk/c3t50060E800382380000008238000000F0d0s0 / ufs 1 no
-
/devices - /devices devfs - no -
ctfs - /system/contract ctfs - no -
objfs - /system/object objfs - no -
swap - /tmp tmpfs - yes -
#
```

18. Modify the `/rtmp/boot/solaris/bootenv.rc` file and change the `setprop bootpath` parameter:

```
# pwd
/rtmp/boot/solaris
# pg bootenv.rc
#
# Copyright 2005 Sun Microsystems, Inc. All rights reserved.
# Use is subject to license terms.
#
#ident "@(#)bootenv.rc 1.32 05/09/01 SMI"
#
# bootenv.rc -- boot "environment variables"
#
setprop kbd-type 'US-English(104-Key)'
setprop ata-dma-enabled '1'
setprop atapi-cd-dma-enabled '0'
setprop ttyb-rts-dtr-off 'false'
setprop ttyb-ignore-cd 'true'
setprop ttya-rts-dtr-off 'false'
setprop ttya-ignore-cd 'true'
setprop ttyb-mode '9600,8,n,1,-'
setprop ttya-mode '9600,8,n,1,-'
setprop lba-access-ok '1'
setprop prealloc-chunk-size '0x2000'
setprop bootpath '/scsi_vhci/disk@g50060e800382380000008238000000f0:a'
#setprop bootpath '/pci@0,0/pci-ide@7/ide@0/cmdk@0,0:a'
setprop console 'text'
```

19. Use the `luxadm probe` command to search for fiber channel devices. Get the device path instance to add in the `bootenv.rc` file:

```
# luxadm probe
Found Fibre Channel device(s):
Node WWN:50060e8003823800 Device Type:Disk device
Logical Path:/dev/rdisk/c3t50060E800382380000008238000000F0d0s2
Node WWN:50060e8003823800 Device Type:Disk device
Logical Path:/dev/rdisk/c3t50060E800382380000008238000000ECd0s2
```

Use the `ls -la` command to examine detailed directory information:

```
# ls -la /dev/rdisk/c3t50060E800382380000008238000000F0d0s0
lrwxrwxrwx 1 root root 68 Jan 12 09:42
/dev/rdisk/c3t50060E800382380000008238000000F0d0s0 ->
../../devices/scsi_vhci/disk@g50060e800382380000008238000000f0:a,raw
```

20. Install the GRUB bootloader to the new root file system:

```
# cd /boot/grub
# /sbin/installgrub stage1 stage2 /dev/rdisk/
c3t50060E800382380000008238000000F0d0s0
stage1 written to partition 0 sector 0 (abs 16065)
stage2 written to partition 0, 232 sectors starting at 50 (abs 16115)
```

21. Update the GRUB bootloader with the bootadm command:

```
# bootadm update-archive -R /rtmp
Creating ram disk on /rtmp
updating /rtmp/platform/i86pc/boot_archive...this may take a minute
```

22. Unmount the file system from the mount point:

```
# umount /rtmp
```

23. Shut down the system to the OK prompt:

```
# sync; halt
```

## Remote SFS SAN Boot Setup with MPxIO Disabled

This section provides the procedure to enable a disk to boot from a StorEdge SFS SAN with MPxIO disabled.

Prerequisites:

- Solaris 10 (x64 and x86) and its latest patches is installed (see the operating system documentation).
- The Emulex SFS device driver for Solaris is installed (see the Sun documentation that accompanies the device driver).
- The Emulex FCA utilities kit is installed.
- x86 BootBIOS is loaded and enabled.
- A SAN disk with formatted capacity at least as large as the local disk.
- The mpzio-disable parameter set to yes in the /kernel/drv/fp.conf file (see the Sun MPxIO documentation).

To set up remote boot, do the following:

1. Start the format utility. At the prompt, type:

```
format
```

2. Select the local drive.

3. Type:

```
partition
```

4. Display the partition layout with the print command. Type:

```
print
```

Leave this window open.

5. Select the LUN you are going to use as the SAN boot disk (not the local drive) using the luxadm probe command. Record this LUN information, which is used throughout this procedure. LUN 0 is used in the example:

```
# luxadm probe
Found Fibre Channel device(s):
  Node WWN:50060e8003823800 Device Type:Disk device
    Logical Path:/dev/rdisk/c2t50060E80038238000000823800d4s2
  Node WWN:50060e8003823800 Device Type:Disk device
    Logical Path:/dev/rdisk/c2t50060E80038238000000823800d3s2
  Node WWN:50060e8003823800 Device Type:Disk device
    Logical Path:/dev/rdisk/c2t50060E8003823800d2s2
```

6. Format the selected LUN with the format command. Specify the path. In the example, the following starts the format utility and specifies the target disk:

```
# format /dev/rdisk/c2t50060E8003823800d0s2
selecting /dev/rdisk/c2t50060E8003823800d0s2
```

[disk formatted]

- After you have formatted the LUN, the Format menu appears. Start the fdisk utility. Type:

```
format> fdisk
```

No fdisk table exists. Accept the default partition that creates:

```
a 100% "SOLARIS System" partition
```

- After the partition is created, results similar to the following appear:

```
Total disk size is 1196 cylinders
Cylinder size is 16065 (512 byte) blocks
Cylinders
Partition  Status  Type           Start  End  Length  %
=====  =====  =====
1         Active  Solaris2      1     1195  1195    100
```

SELECT ONE OF THE FOLLOWING:

- Create a partition
- Specify the active partition
- Delete a partition
- Change between Solaris and Solaris2 Partition IDs
- Exit (update disk configuration and exit)
- Cancel (exit without updating disk configuration)

Enter Selection:

When prompted, select option 5 to exit from disk configuration. Partition 1 of the disk you are going to use as the SAN boot disk is now the active partition and you are still in the format utility.

- The Format menu appears. Prepare the newly created partition for the SAN boot disk. Enter the print command. Type:

```
partition> print
```

```
Current partition table (original):
Total disk cylinders available: 1193 + 2 (reserved cylinders)
Part  Tag      Flag      Cylinders      Size      Blocks
0  unassigned  wm        0              0          (0/0/0)      0
1  unassigned  wm        0              0          (0/0/0)      0
2  backup     wu        0 - 1192      9.14GB     (1193/0/0)  19165545
3  unassigned  wm        0              0          (0/0/0)      0
4  unassigned  wm        0              0          (0/0/0)      0
5  unassigned  wm        0              0          (0/0/0)      0
6  unassigned  wm        0              0          (0/0/0)      0
7  unassigned  wm        0              0          (0/0/0)      0
8  boot       wu        0 - 0         7.84MB     (1/0/0)      16065
9  unassigned  wm        0              0          (0/0/0)      0
```

- Modify the partition of the SAN boot disk with the modify command to match the partition of the source drive that is displayed in the window you opened in step 4:

```
partition> modify
```

Results similar to the following appear:

```
Select partitioning base:
0. Current partition table (original)
1. All Free Hog
Choose base (enter number) [0]? 1
```

At the prompt, enter 1 to start the All Free Hog utility. Note that the All Free Hog utility gives a print out of the current size of the disk. Results similar to the following appear:

```
Part  Tag      Flag      Cylinders      Size      Blocks
0  root     wm        0              0          (0/0/0)      0
```

```

1      swap    wu      0          0          (0/0/0)      0
2      backup  wu      0 - 1192    9.14GB      (1193/0/0) 19165545
3      unassigned  wm      0          0          (0/0/0)      0
4      unassigned  wm      0          0          (0/0/0)      0
5      unassigned  wm      0          0          (0/0/0)      0
6      usr      wm      0          0          (0/0/0)      0
7      unassigned  wm      0          0          (0/0/0)      0
8      boot     wu      0 - 0       7.84MB      (1/0/0)      16065
9      alternates  wm      0          0          (0/0/0)      0

```

Do you wish to continue creating a new partition table based on above table[yes]?

When prompted, type 'yes'. In the example, results similar to the following appear:

```

Enter size of partition '1' [0b, 0c, 0.00mb, 0.00gb]: 2gb
Enter size of partition '3' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '4' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '5' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '6' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '7' [0b, 0c, 0.00mb, 0.00gb]:

```

Part	Tag	Flag	Cylinders	Size	Blocks
0	root	wm	1 - 930	7.12GB	(930/0/0) 14940450
1	swap	wu	931 - 1192	2.01GB	(262/0/0) 4209030
2	backup	wu	0 - 1192	9.14GB	(1193/0/0) 19165545
3	unassigned	wm	0	0	(0/0/0) 0
4	unassigned	wm	0	0	(0/0/0) 0
5	unassigned	wm	0	0	(0/0/0) 0
6	usr	wm	0	0	(0/0/0) 0
7	unassigned	wm	0	0	(0/0/0) 0
8	boot	wu	0 - 0	7.84MB	(1/0/0) 16065
9	alternates	wm	0	0	(0/0/0) 0

Okay to make this the current partition table[yes]?

Accept the current partition table. You are then prompted to label the table. When prompted, enter a table name with quotation marks around it:

```
Enter table name (remember quotes): "disk"
```

The following appears:

```
Ready to label disk, continue? y
```

When prompted, type 'y' to label the disk. Then enter the quit command. Type:

```
partition> quit
```

The Format menu appears. Quit the Format menu. Type:

```
format> quit
```

11. Create a new file system on the root partition that you created on the new SAN boot disk with the newfs command. After results similar to the following appear, type 'y' to construct a new file system:

```
# newfs /dev/rdisk/c2t50060E8003823800d0s0
newfs: construct a new file system /dev/rdisk/c2t50060E8003823800d0s0: (y/n)? y
```

In the example, the following appears:

```
Warning: 13230 sector(s) in last cylinder unallocated
/dev/rdisk/c2t50060E8003823800d0s0:      14940450 sectors in 989
cylinders of 240 tracks, 63 sectors
      7295.1MB in 142 cyl groups (7 c/g, 51.68MB/g, 6272 i/g)
super-block backups (for fsck -F ufs -o b=#) at:
 32, 105936, 211840, 317744, 423648, 529552, 635456, 741360, 847264, 953168,
13979360, 14085264, 14191168, 14297072, 14402976, 14508880, 14614784,
14720688, 14826592, 14932496,
```

12. Create additional file systems as needed.

13. Create a mount point on the local drive and mount the new file system of the boot device:

```
# cd /
# mkdir ptmp
#
# mount /dev/dsk/c2t50060E8003823800d0s0 /ptmp
```

14. Use the `ufsdump` (backup) and `ufsrestore` (restore) commands to copy the contents of the root file system on the source (local) drive to the new file system on the SAN boot disk. In the example, the following results appear:

```
# ufsdump 0f - /dev/rdisk/c1d0s0 | (cd /ptmp; ufsrestore rf -)
DUMP: Date of this level 0 dump: Thu Jan 12 15:01:13:58 2006
DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /dev/rdisk/c1d0s0 (sl2100:/) to standard output.
DUMP: Mapping (Pass I) [regular files]
DUMP: Mapping (Pass II) [directories]
DUMP: Writing 32 Kilobyte records
DUMP: Estimated 6349128 blocks (3100.16MB).
DUMP: Dumping (Pass III) [directories]
DUMP: Dumping (Pass IV) [regular files]
Warning: ./lost+found: File exists
DUMP: 75.03% done, finished in 0:03
DUMP: 6349118 blocks (3100.16MB) on 1 volume at 2976 KB/sec
DUMP: DUMP IS DONE
```

15. Perform a backup and restore on any other file systems created during this process.

16. Change directories to the `/ptmp/etc` directory and modify the `vfstab` file. Replace the local disk instance with the new SAN boot disk instance. In the example, `c1d0s0` is replaced with `c2t50060E8003823800d0s0`:

```
# pwd
/ptmp/etc
# pg vfstab
#device          device          mount          FS          fsck          mount          mount
#to mount        to fsck         point          type        pass         at boot options
#
fd              -          /dev/fd fd      -          no          -
/proc           -          /proc  proc     -          no          -
/dev/dsk/c2t50060E8003823800d0s1
no              -
/dev/dsk/c2t50060E8003823800d0s0
/dev/rdisk/c2t50060E8003823800d0s0      /          ufs          1          no
-
/devices        -          /devices      devfs       -          no          -
ctfs            -          /system/contract      ctfs        -          no          -
objfs           -          /system/object objfs       -          no          -
swap            -          /tmp  tmpfs     -          yes         -
#
```

17. Modify the `/ptmp/boot/solaris/bootenv.rc` file and change the `setprop bootpath` parameter:

```
# pg /ptmp/boot/solaris
bootenv.rc
#
# Copyright 2005 Sun Microsystems, Inc. All rights reserved.
# Use is subject to license terms.
#
#ident "@(#)bootenv.rc 1.32 05/09/01 SMI"
```



```
#
# bootenv.rc -- boot "environment variables"
#
setprop kbd-type 'US-English(104-Key)'
setprop ata-dma-enabled '1'
setprop atapi-cd-dma-enabled '0'
setprop ttyb-rts-dtr-off 'false'
setprop ttyb-ignore-cd 'true'
setprop ttya-rts-dtr-off 'false'
setprop ttya-ignore-cd 'true'
setprop ttyb-mode '9600,8,n,1,-'
setprop ttya-mode '9600,8,n,1,-'
setprop lba-access-ok '1'
setprop prealloc-chunk-size '0x2000'
setprop bootpath '/pci@0,0/pci10de,5d@e/pci10df,fc21@0/fp@0,0/
disk@w50060e8003823800,0:a'
#setprop bootpath '/pci@0,0/pci-ide@7/ide@0/cmdk@0,0:a'
setprop console 'text'
```

18. Use the luxadm probe command to get the device path instance to add in the bootenv.rc file:

```
# luxadm probe
```

Results similar to the following appear:

```
Found Fibre Channel device(s):
Node WWN:50060e8003823800 Device Type:Disk device
Logical Path:/dev/rdisk/c2t50060E8003823800d4s2
Node WWN:50060e8003823800 Device Type:Disk device
Logical Path:/dev/rdisk/c2t50060E8003823800d3s2
Node WWN:50060e8003823800 Device Type:Disk device
Logical Path:/dev/rdisk/c2t50060E8003823800d2s2
Node WWN:50060e8003823800 Device Type:Disk device
Logical Path:/dev/rdisk/c2t50060E8003823800d1s2
Node WWN:50060e8003823800 Device Type:Disk device
Logical Path:/dev/rdisk/c2t50060E8003823800d0s2
```

Use the ls -la command to examine detailed directory information:

```
# ls -la /dev/rdisk/c2t50060E8003823800d0s0
lrwxrwxrwx 1 root root 68 Jan 12 15:04
/dev/rdisk/c2t50060E8003823800d0s0 ->
../../devices/pci@0,0/pci10de,5d@e/pci10df,fc21@0/fp@0,0/
disk@w50060e8003823800,0:a,raw
#
```

19. Install the GRUB bootloader to the new root file system:

```
# cd /boot/grub
# /sbin/installgrub stage1 stage2 /dev/rdisk/c2t50060E8003823800d0s0
stage1 written to partition 0 sector 0 (abs 16065)
stage2 written to partition 0, 232 sectors starting at 50 (abs 16115)
```

20. Update the GRUB bootloader with the bootadm command:

```
# bootadm update-archive -R /ptmp
Creating ram disk on /ptmp
updating /ptmp/platform/i86pc/boot_archive...this may take a minute
```

21. Unmount the file system from the mount point:

```
# umount /ptmp
```

22. Shut down the system to the OK prompt:

```
# sync; halt
```

# OpenBoot

This chapter provides procedures for:

- Configuring OpenBoot (see *Configuring OpenBoot* on page 46)
- Creating Fibre Channel disks (see *Fibre Channel Boot Disks* on page 47)

Troubleshooting information is provided in *Troubleshooting OpenBoot* on page 75.

## Configuring OpenBoot

---

### Configuring OpenBoot for Linux

If you are using Linux on IBM PowerPC (pSeries) with an Emulex driver, refer to the system documentation for instructions on accessing the System Management Services (SMS) feature to set the boot list. SMS is resident in the system firmware.

### Configuring OpenBoot for the Solaris lpfc Driver

To configure any Emulex adapter's boot code for the lpfc driver, type the following at the OBP prompt:

```
ok> show-devs
.
.
.
/pci@5d,700000/lpfc@1
.
.
.
ok> " /pci@5d,700000/lpfc@1" select-dev
ok> set-sd-boot
ok> unselect-dev
ok> reset-all
```

### Configuring OpenBoot for the Solaris SPARC SFS Driver

To configure any Emulex adapter's boot code for the SFS driver, type the following at the OBP prompt:

```
ok> show-devs
.
.
.
/pci@5d,700000/lpfc@1
.
.
.
ok> " /pci@5d,700000/lpfc@1" select-dev
ok> set-sfs-boot
ok> unselect-dev
ok> reset-all
```

### Configuring OpenBoot for AIX

The boot code on AIX systems is referred to as Open Firmware. To install Open Firmware on AIX PCI bus systems, you must use the IBM driver. See your IBM documentation, or visit the IBM Web site for more information.

## Viewing OpenBoot Commands

**Table 2: OpenBoot Commands**

Command	Effect
ok .boot - id	Displays current boot device id
ok .version	Displays current version of OpenBoot
ok .topology	Displays current topology
ok .nvram	Displays current flags for OpenBoot
ok .host-did	Displays actual current AL_PA of adapter
ok .set-default-mode	Resets to default value mode
ok .remove-boot-id	Removes boot id from boot list
ok .devalias	Displays boot list
ok .show-devs	Displays list of devices found
ok .probe-scsi-all	Displays current SCSI and FC devices

## Fibre Channel Boot Disks

---

### Boot from SAN with Linux and Emulex HBAs

If you are using Linux on IBM PowerPC (pSeries), see the system documentation for instructions on accessing the System Management Services (SMS) feature resident in the system firmware to set the boot list.

### Creating a Fibre Channel Remote Boot Disk (Solaris lpfc)

Solaris PCI code is used in the examples. The paths displayed in these procedures are samples. You will enter similar path names, but they will be different from the ones in these sample commands. If you have a Solaris SBus system, change *pci* references to *sbus* and change *lpfc* references to *lpfs*. For example, the command to select the Emulex HBA for a PCI system is:

```
ok" /pci@1f,4000/lpfc@2" select-dev
```

The command for an SBus system is:

```
ok" /sbus@1f,4000/lpfs@2" select-dev
```

From the system monitor prompt:

1. Reset to default value mode. Type:

```
ok" /pci@1f,4000/lpfc@2" select-dev
ok set-default-mode
```

---

**Note:** Always reset the system before making changes to your boot settings. Changes will not be saved if a reset is not performed first.

---

2. Show all system devices. Type:

```
ok show-devs
```

The actual output will vary depending on your system. The path to the Emulex adapter will end with `lpfc@#` or with `fibre-channel`. If the path ends with `fibre-channel`, OpenBoot is not loaded in the adapter's firmware. You must load OpenBoot before continuing with this procedure. This procedure is an example only.

```
/pci@1f,2000/lpfc@1  
/pci@1f,4000/lpfc@2  
/pci@1f,2000/fibre-channel
```

3. Select the Emulex HBA. Select the second HBA shown above by entering the path to the adapter. The path you type will be similar to the one typed in this sample command:

```
ok " /pci@1f,4000/lpfc@2" select-dev
```

4. To check the OpenBoot version, type:

```
ok .version
```

5. To check the topology of the current device, type:

```
ok show-devs  
ok " /pci@1f,4000/lpfc@2" select-dev  
ok .topology  
ok unselect-dev
```

6. To check the link speed of the current device, type:

```
ok show-devs  
ok " /pci@1f,4000/lpfc@2" select-dev  
ok .link-speed  
ok unselect-dev
```

7. To set the link speed of the current device, type:

```
ok show-devs  
ok " /pci@1f,4000/lpfc@2" select-dev  
ok .set-link-speed  
ok unselect-dev
```

OpenBoot supports up to 4 Gigabaud link speed. The default link speed is Auto Select.

8. To set the topology for the intended Fibre Channel boot disk to point-to-point (if the HBA is connected to a switch), type:

```
ok " /pci@1f,4000/lpfc@2" select-dev  
ok set-ptp  
ok unselect-dev  
ok reset-all
```

---

**Note:** Older versions of the OpenBoot PROM (OBP) use "reset". Newer and current versions use "reset-all". If your version requires "reset-all" and you type "reset", nothing will reset.

---

To set the topology back to loop (if the HBA is connected to a hub), type these commands:

```
ok " /pci@1f,4000/lpfc@2" select-dev  
ok set-fc-al  
ok unselect-dev  
ok reset-all
```

By default, the topology is set to auto-FCAL. To set auto topology with public or private loop first, type these commands:

```
ok " /pci@1f,4000/lpfc@2" select-dev
ok set-auto-fcal
ok unselect-dev
ok reset-all
```

To set auto topology with point-to-point first, type these commands:

```
ok " /pci@1f,4000/lpfc@2" select-dev
ok set-auto-ptp
ok unselect-dev
ok reset-all
```

9. To view the all SCSI and Fibre Channel devices, type:

```
ok probe-scsi-all
```

10. To view the all WWPNs and LUNs, type:

```
ok show-children
```

11. To view the desired WWPN and LUNs, type:

```
ok 2200002037997f70 show-wwpn-child
```

12. To view the current boot device ID, type:

```
ok show-devs
ok "/pci@1f,0/pci@1/lpfc@1" select-dev
/* select lpfc@1 (for example) */
ok .boot-id
```

Make note of the actual wwpn, did or alpa returned from the probe and write down the corresponding boot entry.

13. To set the boot device ID, type:

```
ok "/pci@1f,0/lpfc@1" select-dev
ok [wwpn|did|alpa] [lun] [target id] set-boot-id
ok unselect-dev
```

where [wwpn|did|alpa] - enter the device WWPN, DID, or AL\_PA of the storage device.

[lun] - enter the LUN number in hexadecimal. To enter the decimal LUN number, enter d# [lun].

[target id] - enter the target ID in hexadecimal. To enter the decimal target ID, enter d# [target id].

---

**Note:** It is recommended to use the WWPN in most cases. The DID and AL\_PA may change between boots, causing the SAN boot to fail, unless the DID and AL\_PA are specifically configured to not change between boots.

---

Example 1: alpa=e1, lun=100 (decimal) and target id=10 (decimal):

```
ok alpa e1 d# 100 d# 10 set-boot-id
```

Example 2: wwpn=50000034987AFE, lun=af (hexadecimal) and target id=10 (decimal):

```
ok wwpn 50000034987AFE af d# 10 set-boot-id
```

Example 3: did=6312200, lun=25 (hexadecimal) and target id=f (hexadecimal):

```
ok did 6312200 25 f set-boot-id
```

14. To remove the boot device ID, type:

```
ok "/pci@1f,0/pci@1/lpfc@1" select-dev
ok remove-boot-id /* to clear boot id settings */
ok unselect-dev
```

or

```
ok "/pci@1f,0/pci@1/lpfc@1" select-dev
```

```
ok set-default-mode      /* to clear boot id settings */
ok unselect-dev
```

15. To switch between SFS and lpfc (SD) boot format, type:

```
ok set-sfs-boot          /* switch to SFS Boot Format*/
ok set-sd-boot           /* switch to SD Boot Format*/
```

16. To set the FCode diag switch, type:

```
ok set-fdiag-switch      /* print debug message during POST */
```

If the FCode diag switch is on, the detail debug error message prints out during POST or at boot time if there is a link or boot failure.

17. Boot to the original local disk to set up the newly defined Fibre Channel disk. Type:

```
ok boot [local disk]
```

where [local disk] - enter the complete path or the alias of the original boot disk.

---

**Note:** If the Fibre Channel disk is not found when the system is rebooted, enter "touch / reconfigure" and reboot. If the disk is still not discovered, edit the "/kernel/drv/lpfc.conf" file and change the topology from 4 to 2 or reverse. It may also be necessary to add an entry for the boot drive to the sd.conf file.

---

18. To display all of the current flags for OpenBoot, type:

```
ok show-devs
ok "/pci@1f,0/pci@1/lpfc@1" select-dev
/* select lpfc@1 (for example) */
ok .nvram
```

Flag information will be displayed (see Table 3).

**Table 3: Signature Table**

Signature	OpenBoot signature
Valid_flag	internal flag for OpenBoot
Host_did	shows host DID number
Enable_flag	internal flag for OpenBoot
Topology_flag	topology flag for OpenBoot
Link_Speed_Flag	set link speed
Diag_Switch	set fcode diag switch
Boot_id	shows target ID number
Lnk_timer	internal use for OpenBoot
Plogi_timer	internal use for OpenBoot
LUN	shows boot LUN in use
DID	shows boot ID in use
WWPN	shows boot WWPN in use

---

**Note:** Target ID can be bound to either DID (destination ID) or WWPN (worldwide port name) and saved in an adapter nvram. It can also be saved in an environmental variable boot-device.

---

### Making a Bootable System Disk (Solaris lpfc Driver)

The examples in this procedure use the logical drive designated as "c1t1d0" (controller 1, target 1, lun 0). Your logical drive may be different.

1. Boot from the local system disk.
2. Run the format utility. Type:
 

```
# format
```
3. Select the logical drive to become the boot disk (in this case, c1t1d0).

4. Select the partition option and partition the disk as desired.
5. Select the label option and write a volume label to the target disk.  
For help with the format utility, refer to the man page "man format".
6. Install the boot on partition 0 of the target disk. (Type this command as one line.)  
# installboot /usr/platform/ `uname -i`/lib/fs/ufs/bootblk /dev/rdisk/clt1d0s0
7. Create a filesystem for each partition that will contain a mounted filesystem. Type:  
# newfs -v /dev/rdisk/clt1d0s0 (will become root)  
# newfs -v /dev/rdisk/clt1d0s6 (will become usr)  
# newfs -v /dev/rdisk/clt1d0s7 (will become export/home)
8. Create temporary mount points for the new partitions. Type:  
# mkdir root2  
# mkdir usr2  
# mkdir export2
9. /dev/fd must be unmounted to copy the root file system. First, kill the volume manager. Type:  
# ps -ef|grep vold (to obtain the PID or process id)  
# kill -9 PID (where PID is obtained above)  
# umount /dev/fd
10. Mount, copy, then unmount the usr2 file system. Type:  
# mount /dev/dsk/clt1d0s6 /usr2  
# cd /usr  
# tar cvf - \* | ( cd /usr2; tar xf -)  
# umount /usr2
11. Copy the export/home file system. Type:  
# mount /dev/dsk/clt1d0s7 /export2  
# cd /export# tar cvf - \* | (cd /export2; tar xf -)  
# umount /export2
12. Copy the root file system by first building a file list. Type:  
# cd /  
ls -a > root\_copy
13. Use the vi editor to edit root\_copy and delete all entries that refer to mount points.  
., ., usr, tmp, proc, export, root2, export2, usr2, opt (if a separate mount point)  
and any other apparent filesystems displayed by the df command. Your goal in editing this file is to have the root\_copy point to system files on the root directory without pointing to mount points.
14. Join file names so that they are on one line; if you are using vi, press <shift> <j> to do this.
15. Add the tar command to the file list so that the result looks like this:  
tar cvf - .Xauthority .cpr\_config .dt .dtprofile TT\_DB  
arg bak bin cdrom dev devices doit etc firmware fun kernel lib mnt mt net opt  
platform sbin timer tmp var vol xfn
16. Perform copy. Type:  
# mount /dev/dsk/clt1d0s0 /root2  
chmod 777 root\_copy  
sh root\_copy
17. Create the additional mount points (filesystem names). Type:  
# cd /root2  
# mkdir export  
# mkdir opt (if opt is a separate mount point)

```
# mkdir proc# mkdir tmp
# mkdir vol
# mkdir usr
```

18. Repeat for any other mounts unique to your system that were excluded in the copy.
19. Edit `/root2/etc/vfstab` changing the controller number, target number, and LUN number to point to the new Fibre Channel boot disk. For example, if the Fibre Channel boot disk is `c1t1d0`, replace all local disk entries of `c0t0d0` with `c1t1d0`.

Currently file shows:
<pre>/dev/dsk/c0t0d0s1 (swap)  /dev/dsk/c0t0d0s0 and /dev/rdisk/c0t0d0s0 (root) /dev/dsk/c0t0d0s6 and /dev/rdisk/c0t0d0s6 (usr) /dev/dsk/c0t0d0s7 and /dev/rdisk/c0t0d0s7 (export)</pre>

Edit file to show:
<pre>/dev/dsk/clt1d1s1 (swap)  /dev/dsk/clt1d0s0 and /dev/rdisk/clt1d0s1 (root) /dev/dsk/clt1d0s6 and /dev/rdisk/clt1d0s6 (usr) /dev/dsk/clt1d0s7 and /dev/rdisk/clt1d0s7 (export)</pre>

20. Reboot the system. Type:

```
# sync
# sync
# halt
ok reset-all
```

21. Boot to disk. Type:

```
ok boot disk
```

The system should boot to the Fibre Channel disk.

22. View the current dump device setting. Type:

```
# dumpadm
```

23. Change the dump device to the swap area of the Fibre Channel drive. Type:

```
# dumpadm -d /dev/dsk/clt1d0s1
```

where `/dev/dsk/clt1d0s1` is a sample path to the swap area of the Fibre Channel drive. Your path may be different from the one shown here. Enter your path.

## Creating a Fibre Channel Remote Boot Disk (Solaris SPARC SFS Driver)

This section provides the procedure for enabling a disk to boot from a StorEdge SFS SAN on a SPARC system.

Prerequisites:

- Solaris 8 (32- and 64-bit), Solaris 9 (32- and 64-bit) or Solaris 10 (32- and 64-bit) for SPARC and its latest patches is installed (see the operating system documentation).
- The Emulex SFS device driver for Solaris is installed (see the Sun documentation that accompanies the device driver).
- The Emulex FCA utilities kit is installed.
- Boot code (OpenBoot) is installed.
- A SAN disk with formatted capacity at least as large as the local disk.



- The `mpxio-disable` parameter is set to `yes` in the `/kernel/drv/fp.conf` file (see the Sun MPxIO documentation).
- For Solaris 8 or 9, the `mpxio-disable` parameter is set to `yes` in the `/kernel/drv/scsi_vhci.conf` (see the Sun MPxIO documentation).

To set up remote boot, do the following:

1. At the prompt, type:

```
format
```

2. Select the local drive.

3. Type:

```
partition
```

4. To display the partition layout, type:

```
print
```

Leave this window open.

5. Run the format utility:

```
#format
```

```
Searching for disks...done
```

```
.
```

```
.
```

```
31. c5t226000C0FF9833AFd6 <DotHill-SANnetIIFC-327R cyl 9698 alt 2 hd 64 sec 32
    /pci@780/pci@0/pci@8/SUNW,emlxs@0/fp@0,0/ssd@w226000c0ff9833af,6
```

```
Specify disk (enter its number):
```

6. Enter the device you are going to use as the SAN boot disk (this must not be the local drive), and record its device path so that you can use it in step 20.

7. The Format menu appears. To prepare the partition for the SAN boot disk, type:

```
partition
```

8. The Partition menu appears. To display the current partition table, type:

```
print
```

9. The current partition table appears:

Part	Tag	Flag	Cylinders	Size	Blocks
0	root	wm	0 - 127	128.00MB	(128/0/0) 262144
1	swap	wu	128 - 255	128.00MB	(128/0/0) 262144
2	backup	wu	0 - 9697	9.47GB	(9698/0/0) 19861504
3	unassigned	wm	0	0	(0/0/0) 0
4	unassigned	wm	0	0	(0/0/0) 0
5	unassigned	wm	0	0	(0/0/0) 0
6	usr	wm	256 - 9697	9.22GB	(9442/0/0) 19337216
7	unassigned	wm	0	0	(0/0/0) 0

```
partition>
```

10. Modify the partition of the SAN boot disk to match the partition of the local drive that is displayed in the window you opened in step 4:

```
#modify
```

```
Select partitioning base:
```

```
0. Current partition table (original)
```

```
1. All Free Hog
```

```
Choose base (enter number) [0]? 1
```

Part	Tag	Flag	Cylinders	Size	Blocks
0	root	wm	0	0	(0/0/0) 0
1	swap	wu	0	0	(0/0/0) 0
2	backup	wu	0 - 9697	9.47GB	(9698/0/0) 19861504
3	unassigned	wm	0	0	(0/0/0) 0
4	unassigned	wm	0	0	(0/0/0) 0
5	unassigned	wm	0	0	(0/0/0) 0
6	usr	wm	0	0	(0/0/0) 0
7	unassigned	wm	0	0	(0/0/0) 0

Do you wish to continue creating a new partition table based on above table[yes]?

Free Hog partition[6]? 0

Enter size of partition '1' [0b, 0c, 0.00mb, 0.00gb]: 4gb

Enter size of partition '3' [0b, 0c, 0.00mb, 0.00gb]: 0

Enter size of partition '4' [0b, 0c, 0.00mb, 0.00gb]: 0

Enter size of partition '5' [0b, 0c, 0.00mb, 0.00gb]: 0

Enter size of partition '6' [0b, 0c, 0.00mb, 0.00gb]: 0

Enter size of partition '7' [0b, 0c, 0.00mb, 0.00gb]: 0

Part	Tag	Flag	Cylinders	Size	Blocks
0	root	wm	0 - 5601	5.47GB	(5602/0/0) 11472896
1	swap	wu	5602 - 9697	4.00GB	(4096/0/0) 8388608
2	backup	wu	0 - 9697	9.47GB	(9698/0/0) 19861504
3	unassigned	wm	0	0	(0/0/0) 0
4	unassigned	wm	0	0	(0/0/0) 0
5	unassigned	wm	0	0	(0/0/0) 0
6	usr	wm	0	0	(0/0/0) 0
7	unassigned	wm	0	0	(0/0/0) 0

Okay to make this the current partition table[yes]?

Enter table name (remember quotes): "disk"

Ready to label disk, continue? yes

partition>

- Quit the Format utility and create a new file system on the root partition that you created on the new SAN boot disk:

```
# newfs /dev/rdisk/c5t226000C0FF9833AFd6s0
newfs: construct a new file system /dev/rdisk/c5t226000C0FF9833AFd6s0: (y/n)? y
/dev/rdisk/c5t226000C0FF9833AFd6s0:      11472896 sectors in 5602 cylinders of
64 tracks, 32 sectors
      5602.0MB in 115 cyl groups (49 c/g, 49.00MB/g, 6144 i/g)
super-block backups (for fsck -F ufs -o b=#) at:
 32, 100416, 200800, 301184, 401568, 501952, 602336, 702720, 803104, 903488,
10538304, 10638688, 10739072, 10839456, 10939840, 11040224, 11140608,
11240992, 11341376, 11441760,
#
```

- Create additional file systems as needed.

- Create a mount point on the local drive and mount the new file system of the boot device:

```
#cd
#mkdir ztmp
#mount /dev/dsk/c5t226000C0FF9833AFd6s0 /ztmp
#
```

14. Install the boot block using the `installboot` command, for example:

```
# /usr/sbin/installboot /usr/platform/`uname -i`/lib/fs/ufs/bootblk /dev/rdisk/
c5t226000C0FF9833AFd6s0
```

15. Use the `ufsdump` (backup) and `ufsrestore` (restore) commands to copy the contents of the root file system on the local drive to the new file system on the SAN boot disk:

```
ufsdump 0f - /dev/rdisk/c2t0d0s0 | (cd /ztmp; ufsrestore rf -)

# ufsdump 0f - /dev/rdisk/c2t0d0s0 | (cd /ztmp; ufsrestore rf -)
DUMP: Date of this level 0 dump: Fri Dec 09 10:59:05 2005
DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /dev/rdisk/c2t0d0s0 (dhcp-91-202.emulex.com:/) to standard out-
put.
DUMP: Mapping (Pass I) [regular files]
DUMP: Mapping (Pass II) [directories]
DUMP: Writing 32 Kilobyte records
DUMP: Estimated 7299642 blocks (3564.28MB).
DUMP: Dumping (Pass III) [directories]
DUMP: Dumping (Pass IV) [regular files]
Warning: ./lost+found: File exists
DUMP: 7299582 blocks (3564.25MB) on 1 volume at 10430 KB/sec
DUMP: DUMP IS DONE
#
```

16. Perform a backup and restore on any other file systems that were created during this process.

17. Change directories to the `/ztmp/etc` directory and modify the `vfstab` file. Replace the local disk instance with the new SAN boot disk instance. In the following example, `c2t0d0` is replaced with `c5t226000C0FF9833AFd6`:

```
# pg vfstab
#device          device          mount          FS          fsck          mount          mount
#to mount        to fsck         point          type         pass         at boot      options
#
fd              -              /dev/fd fd     -            -              -            -
/proc           -              /proc  proc        -            -              -            -
/dev/dsk/c5t226000C0FF9833AFd6s1 -              -              -            swap          -            no
/dev/dsk/c5t226000C0FF9833AFd6s0 /dev/rdisk/c5t226000C0FF9833AFd6s0
/               ufs            1              no           -              -            -
/devices        -              /devices      devfs        -            no           -
ctfs            -              /system/contract ctfs         -            no           -
objfs           -              /system/object objfs        -            no           -
swap            -              /tmp  tmpfs       -            yes           -
```

18. Unmount the file system from the mount point:

```
# umount /ztmp
```

19. Shut down the system to the OK prompt.

20. Using the device path that you recorded in step 6, but with `disk` replacing `ssd`, boot into the new SAN boot disk with the `-r` option (in the example, the device path is `/pci@780/pci@0/pci@8/SUNW,emlxs@0/fp@0,0/disk@w226000c0ff9833af,6`):

```
{0} ok boot /pci@780/pci@0/pci@8/SUNW,emlxs@0/fp@0,0/disk@w226000c0ff9833af,6 -
r
```

21. Log in and verify that you have booted from the SAN boot disk. Use the `df -k` command to see the new file system:

```
# df -k
Filesystem          kbytes  used  avail capacity  Mounted on
/dev/dsk/c5t226000C0FF9833AFd6s0
```

```

5646270 3556560 2033248 64% /
/devices 0 0 0 0% /devices
ctfs 0 0 0 0% /system/contract
proc 0 0 0 0% /proc
mnttab 0 0 0 0% /etc/mnttab
swap 10938792 1040 10937752 1% /etc/svc/volatile
objfs 0 0 0 0% /system/object
/platform/sun4v/lib/libc_psr/libc_psr_hwcap1.so.1
5646270 3556560 2033248 64% /platform/sun4v/lib/
libc_psr.so.1
/platform/sun4v/lib/sparcv9/libc_psr/libc_psr_hwcap1.so.1
5646270 3556560 2033248 64% /platform/sun4v/lib/
sparcv9/libc_psr.so.1
fd 0 0 0 0% /dev/fd
swap 10937752 0 10937752 0% /tmp
swap 10937776 24 10937752 1% /var/run

```

22. Configure the system crash dump for the new SAN boot device on slice 1, using the `dumpadm` command with the `-d` option:

```

# dumpadm -d /dev/dsk/c5t226000C0FF9833AFd6s1
    Dump content: kernel pages
    Dump device: /dev/dsk/c5t226000C0FF9833AFd6s1 (swap)
Savecore directory: /var/crash/dhcp-91-202.emulex.com
Savecore enabled: yes
#

```

## EFIBoot

This chapter provides procedures for:

- Configuring EFIBoot (see *Configuring EFIBoot* on page 57)

---

**Note:** This section reflects the most recent release of the EFI utility. Some selections may not be available if you are using an older version of the utility.

---

- Creating a Fibre Channel boot disk (see *Installing the Server* on page 67)

In addition, an EFIBoot reference section is provided (see *EFIBoot Reference* on page 68).

Troubleshooting information is provided in *Troubleshooting EFIBoot* on page 75.

## Configuring EFIBoot

---

The EFI utility has many options that can be modified to provide for different behavior. Use the EFI utility to do the following tasks:

- Enable the HBA to boot from SAN
- Change the default AL\_PA
- Change the topology
- Change the PLOGI retry timer
- Change the device path
- Change the maximum LUNs per target
- Set up a boot target scan
- Configure a boot device
- Cancel a boot device
- Update EFIBoot

## Accessing the EFI Utility

---

**Note:** Before you can use the EFI utility for the first time, you must install it. See *Installing the EFI Utility* on page 69 for instructions.

---

To access the main EFI utility:

1. To view Emulex driver handle information, at the shell prompt type:

```
fs0:> drivers
```

A list of drivers is displayed (Figure 40).

DRIVER	VERSION	TYPE	CONF	NO	HC	DRIVER NAME	IMAGE NAME
10	00000010	B	- -	4	10	PCI Bus Driver	PciBus
14	00000029	D	- -	2	-	Usb Ohci Driver	UsbOhci
15	00000010	B	- -	2	2	USB Bus Driver	UsbBus
16	00001010	D	X -	1	-	Usb Keyboard Driver	UsbKb
17	00000010	?	- -	-	-	Usb Mouse Driver	UsbMouse
18	00000010	D	- -	1	-	Usb Bot Mass Storage Driver	UsbBot
19	00000010	?	- -	-	-	Usb Cbi0 Mass Storage Driver	UsbCbi0
1A	00000010	?	- -	-	-	<UNKNOWN>	UsbCbi1
1B	00000010	D	- -	1	-	Generic USB Mass Storage Driver	UsbMassStorage
1C	00000010	?	- -	-	-	UGA Console Driver	GraphicsConsole
1D	00000000	D	- -	1	-	PCI VGA Mini Port Driver	PciVgaMiniPort
1E	00000010	D	- -	1	-	VGA Class Driver	VgaClassDriver
1F	00000010	B	- -	1	1	Serial 16550 UART Driver	Serial16550
20	00000010	B	- -	1	1	Serial Terminal Driver	Terminal
21	00000010	D	- -	2	-	Platform Console Management Driver	ConPlatform
22	00000010	D	- -	2	-	Platform Console Management Driver	ConPlatform
23	00000010	B	- -	2	2	Console Splitter Driver	ConSplitter
86	00031114	D	X X	1	-	Emulex SCSI Pass Thru Driver	\\elxcli311a4.efi

Figure 40: Driver Information Example

2. Enter the following command followed by the device handle for the SCSI Pass Thru Driver, for example, 86 (your device handle number may be different):

```
drvcfg-s 86
```

3. Press <Enter>. Information similar to Table 4 is displayed, listing all of the HBAs in the system. Your listing will vary depending on the boot package you are using.

Table 4: HBA Information Example

001	LP1000DC	PCI-X	100 MHZ	BUS#: 40	Dev#: 01	Func# 00
002	LP8000	PCI	66 MHZ	BUS#: 61	Dev#: 02	Func# 00
003	LP8000	PCI	66 MHZ	BUS#: 61	Dev#: 02	Func# 00
004	LP952	PCI-X	100 MHZ	BUS#: A0	Dev#: 01	Func# 00
005	LP850	PCI	66 MHZ	BUS#: C0	Dev#: 02	Func# 00
006	LP982	PCI-X	100 MHZ	BUS#: C0	Dev#: 01	Func# 00
007	LP9802	PCI-X	100 MHZ	BUS#: E1	Dev#: 01	Func# 00
008	LP9802	PCI-X	100 MHZ	BUS#: E1	Dev#: 01	Func# 00

4. Locate the HBA that you want to configure and select (highlight) the row and press <Enter>. The EFI utility menu is displayed (Figure 41).

```

001: LP1000DC PCI-X 66MHz Bus#: 80 Dev#: 03 Func#: 01
      HBA Status: Ready      Boot Bios : Disabled
      EXP ROM   : Disabled    Firmware  : TS1.91A1
  
```

---

1. Setup Utility
2. Firmware Update

<↑/↓> to Change Options. <Enter> to Select Option. <ESC> to Prev Menu.

Figure 41: EFI Utility Main Menu

## Conventions for the EFI Utility

The EFI utility contains menu windows and configuration windows. A navigation bar is located at the bottom of the each window:

- Press the up and down arrows on your keyboard to move through the menu options or configuration fields.
- Press the left and right arrows on your keyboard to scroll through pages of information.
- Press the <Enter> key to select a menu option, to select a row in a configuration window, or to change a configuration default.
- Usually press the <Esc> key to return to the previous menu. There are a few exceptions to this convention.

## Making Configuration Changes Effective

1. After you have made all configuration changes for the HBAs in your system, exit the EFI utility.
2. In an EFI shell, enter the following command:

```
connect -r
```

This command will make configuration changes effective for all HBAs in your system.

## Configuration Tasks

### Enabling the HBA to Boot from SAN

To enable the HBA to boot from SAN:

1. Access the EFI utility (see page 57), select 1, Setup Utility and press <Enter>. The following menu is displayed:

- |  |
|--|
| <ol style="list-style-type: none"> <li>1. Enable or Disable BIOS</li> <li>2. Configure HBA Parameters</li> <li>3. Configure Boot Parameters</li> <li>4. Reset to Default Values</li> </ol> |
|--|

Figure 42: Setup Utility Menu

- Select **1**, Enable or Disable BIOS and press <Enter>. The following window is displayed:

```

EFI-BIOS Boot Option is currently DISABLED.

Disabled - default
Enabled:
  
```

Figure 43: BIOS Disabled

- Select Enabled and press <Enter>. The following window is displayed:

```

EFI-BIOS Boot Option is currently Enabled.

Disabled - default
Enabled
  
```

Figure 44: BIOS Enabled

---

**Note:** Press <Esc> to return to the EFI utility menu.

---

## Changing the Default AL\_PA

When a device is connected to a Fibre Channel arbitrated loop, the loop initialization process (LIP) automatically assigns an AL\_PA (arbitrated loop physical address) to the device. This number is assigned to the device as long as it is connected to the loop and powered up. When the device is removed from the loop or turned off, as it is reconnected or when another LIP occurs, the device may be assigned a different AL\_PA. The HardALPA=0xn parameter allows you to permanently assign an AL\_PA to a host adapter. This does not alter the host bus adapter flash ROM. The default for this parameter is 0x00. All valid AL\_PA values are listed in Table 5.

**Table 5: Valid AL\_PA Values**

0x00	0x01	0x02	0x04	0x08	0x0F	0x10	0x17
0x18	0x1B	0x1D	0x1E	0x1F	0x23	0x25	0x26
0x27	0x29	0x2A	0x2B	0x2C	0x2D	0x2E	0x31
0x32	0x33	0x34	0x35	0x36	0x39	0x3A	0x3C
0x43	0x45	0x46	0x47	0x49	0x4A	0x4B	0x4C
0x4D	0x4E	0x51	0x52	0x53	0x54	0x55	0x56
0x59	0x5A	0x5C	0x63	0x65	0x66	0x67	0x69
0x6A	0x6B	0x6C	0x6D	0x6E	0x71	0x72	0x73
0x74	0x75	0x76	0x79	0x7A	0x7C	0x80	0x81
0x82	0x84	0x88	0x8F	0x90	0x97	0x98	0x9B
0x9D	0x9E	0x9F	0xA3	0xA5	0xA6	0xA7	0xA9
0xAA	0xAB	0xAC	0xAD	0xAE	0xB1	0xB2	0xB3
0xB4	0xB5	0xB6	0xB9	0xBA	0xBC	0xC3	0xC5
0xC6	0xC7	0xC9	0xCA	0xCB	0xCC	0xCD	0xCE
0xD1	0xD2	0xD3	0xD4	0xD5	0xD6	0xD9	0xDA
0xDC	0xE0	0xE1	0xE2	0xE4	0xE8	0xEF	



To change the default AL\_PA:

1. Access the EFI utility (page 57), select **1**, Setup Utility and press <Enter>. The setup utility menu is displayed (Figure 42 on page 59).
2. Select **2**, Configure HBA Parameters and press <Enter>. The following menu is displayed:

```
1. Default ALPA of this adapter
2. Topology Selection
3. PLOGI Retry Timer
```

Figure 45: HBA Configuration Menu

3. Select **1**, Default ALPA of this adapter and press <Enter>. The following message is displayed:

```
Requested ALPA value stored in NVRAM: 0x00
HardALPA: 0x004
```

Figure 46: Default AL\_PA Setup

4. Select the HardALPA value field and use the up and down arrows to change the default value. The new value is displayed, for example.

```
Requested ALPA value stored in NVRAM: 0x1F
HardALPA: 0x1F
```

Figure 47: New Default AL\_PA Setup

---

**Note:** Press <Esc> to return to the EFI utility menu.

---

## Changing the Topology

Emulex drivers support arbitrated loop and point-to-point topologies.

To change the topology:

1. Access the EFI utility (page 57), select **1**, Setup Utility and press <Enter>. The Setup utility menu is displayed (Figure 42 on page 59).
2. Select **2**, Configure HBA Parameters and press <Enter>. The HBA Configuration menu is displayed (Figure 45 on page 61).
3. Select **2**, Topology Selection and press <Enter>. The following window is displayed:

```
Current Topology: Auto (Start FC-AL)
Auto (Start FC-AL) - Default
Auto (Start Point-to-Point)
Point to Point
FC-AL
```

Figure 48: Topology Options with Default Topology

4. Use the up and down arrows to select a different topology. The window is displayed with the new value, for example:

```
Current Topology: Point-to_Point
Auto (Start FC-AL) - Default
Auto (Start Point-to-Point)
Point to Point
FC-AL
```

Figure 49: Topology Options with Point-to-Point Selected

---

**Note:** The presence of a fabric is detected automatically.

---

## Changing the PLOGI Retry Timer

This option allows you to set the interval for the PLOGI (port log in) retry timer. This option is especially useful for Tachyon-based RAID arrays. Under very rare occasions, a Tachyon-based RAID array will reset itself and the port will go offline temporarily in the loop. When the port comes to life, the PLOGI retry interval scans the loop to discover this device.

To change timer values:

1. Access the EFI utility (page 57), select **1**, Setup Utility and press <Enter>. The setup utility menu is displayed (Figure 42 on page 59).
2. Select **2**, Configure HBA Parameters and press <Enter>. The HBA configuration menu is displayed (Figure 45 on page 61).
3. Select **3**, PLOGI Retry Timer and press <Enter>. The following window is displayed:

```

PLOGI retry time is: Disabled - No Retry
                    Disabled- Default
                    50 msec
                    100 msec
                    200 msec
    
```

Figure 50: PLOGI Retry Timer Setup with the Default

4. Use the up and down arrows to select a retry timer option. The window is displayed with the new value, for example:

```

PLOGI retry time is: 50 msec
                    Disabled- Default
                    50 msec
                    100 msec
                    200 msec
    
```

Figure 51: PLOGI Retry Timer Setup with 50 msec

---

**Note:** Press <Esc> to return to the EFI utility menu.

---

## Changing the Device Path

To change the device path:

1. Access the EFI utility (page 57), select **1**, Setup Utility and press <Enter>. The setup utility menu is displayed (Figure 42 on page 59).
2. Select **1**, Configure Boot Parameters and press <Enter>. The boot parameter configuration menu is displayed:

```

1. Device Path Selection
2. Maximum Luns/Target
3. Boot Target Scan
4. Configure Boot Devices
5. Delay Device Discovery
    
```

Figure 52: Boot Parameter Configuration Menu

3. Select **1**, Device Path Selection and press <Enter>. The following window is displayed:

```

Device path set to Fibre - Default
                    SCSI
    
```

Figure 53: Device Path, Default Selected

- Use the up and down arrows to change the device path. The window is displayed with the new value, for example:

Device path set to SCSI  
 Fibre - Default

Figure 54: Device Path, SCSI Selected

---

**Note:** The device path is typically not changed. The SCSI device path is used for legacy systems.

---

## Changing the Maximum LUN per Target Setting

The maximum number of LUNs represents the maximum number of LUNs that will be polled during device discovery. The minimum value is 1, the maximum value is 4096. The default is 256.

To change the maximum number of LUNs:

- Access the EFI utility (page 57), select **1**, Setup Utility and press <Enter>. The setup utility menu is displayed (Figure 42 on page 59).
- Select **3**, Configure Boot Parameters and press <Enter>. The boot parameter configuration menu is displayed (Figure 52 on page 62).
- Select **2**, Maximum LUNs/Target and press <Enter>. A window is displayed with the current maximum LUNs value.

Current Maximum LUNs Support is: 0256 decimal  
 Maximum LUNs/Target: 0256

Figure 55: Current Maximum LUNs with the Default Setup

- To change the maximum LUN value, use the up and down arrows to change the value (between 1 and 4096). Press <Enter>. A window is displayed with the new value. For example:

Current Maximum LUNs Support is: 0080 decimal  
 Maximum LUNs/Target: 0080

Figure 56: Current Maximum LUNs with 0080 as the Setup

---

**Note:** 256 is the default and typical number of LUNs in a device.

---



---

**Note:** Press any key to return to the configuration menu.

---

## Setting Up Boot Target Scan

This option is available only if none of the eight boot entries are configured to boot via DID or WWPN. It is strongly recommended that you use the Configure Boot Devices menu to configure eight boot entries for fabric point-to-point, public loop or private loop configurations.

With boot scan enabled, the first device will issue a Name Server Inquiry.

To set up a boot target scan:

- Access the EFI utility (page 57), select **1**, Setup Utility and press <Enter>. The setup utility menu is displayed (Figure 42 on page 59).
- Select **3**, Configure Boot Parameters and press <Enter>. The boot parameter configuration menu is displayed (Figure 52 on page 62).

3. Select **3**, Boot Target Scan and press <Enter>. The following window is displayed:

```

Current Boot Scan Setting is NVRAM Targets
Boot Path from NVRAM Targets (Default)
Boot Path Discovered Targets
Do not Create Boot Path
  
```

Figure 57: Boot Scan Options with the Default Selected

4. Use the up and down arrows to select an option and press <Enter>. The new boot scan setting is at the top of the window. For example:

```

Current Boot Scan Setting is Discovered Targets
Boot Path from NVRAM Targets (Default)
Boot Path Discovered Targets
Do not Create Boot Path
  
```

Figure 58: Boot Scan Options with Discovered Targets Selected

---

**Note:** Press <Esc> to return to the EFI utility menu.

---

## Configuring a Boot Device

To configure a boot device:

1. Access the EFI utility (page 57), select **1**, Setup Utility and press <Enter>. The setup utility menu is displayed (Figure 42 on page 59).
2. Select **3**, Configure Boot Parameters and press <Enter>. The boot parameter configuration menu is displayed (Figure 52 on page 62).
3. Select **4**, Configure Boot Devices and press <Enter>. Information similar to Figure 59 is displayed:

01	DID:000000	WWN:00000000	00000000
02	DID:000000	WWN:00000000	00000000
03	DID:000000	WWN:00000000	00000000
04	DID:000000	WWN:00000000	00000000
05	DID:000000	WWN:00000000	00000000
06	DID:000000	WWN:00000000	00000000
07	DID:000000	WWN:00000000	00000000
08	DID:000000	WWN:00000000	00000000

Figure 59: Device Row Example

4. Select a device row and press <Enter>. The following window is displayed. The device row selected in the previous step is displayed in this window, for example:

```

Selected Boot Device in Flash:
01          DID:000000    WWN:00000000  00000000    LUN: 0000
1. Scan Targets
2. Clear Flash Boot Info
  
```

Figure 60: Boot Device Configuration Options

---

**Note:** Press the <Esc> key to return to the list of devices.

---

- Select **1**, Scan Targets and press <Enter>. The system scans the HBA and shows your targets. Figure 61 shows one target:

001	DID:011700	WWN:50060161	106009CA
-----	------------	--------------	----------

Figure 61: Target Listing Example

---

**Note:** Eight targets are displayed per window. If more than eight targets are configured, press the key (right arrow) to view additional targets eight at a time.

---

- Select a target row and press <Enter>. LUNS for the target is displayed. Figure 62 shows LUNS 0000 through 0003:

0001	Mode: Peripheral dev	LUN: 0000
0002	Mode: Peripheral dev	LUN: 0001
0003	Mode: Peripheral dev	LUN: 0002
0004	Mode: Peripheral dev	LUN: 0003

Figure 62: LUN Listing Example

---

**Note:** Eight LUNs are displayed per window. If more than eight LUNs are configured for the target, press the right arrow to view more LUNs.

---

- For this example, select LUN: 0001 and press <Enter>. The following menu is displayed:

Selected Boot Device:				
001	DID:011700	WWN:50060161	106009CA	001
1. Boot this device via WWN.				
2. Boot this device via DID.				

Figure 63: LUN Boot Options

---

**Note:** Press <Esc> to return to the list of devices.

---

- Select **1**, Boot this device via WWN and press <Enter>. Figure 64 shows the LUN selected in the previous step.

01	DID:000000	WWN:50060161	106009CA	Mode: Per LUN: 0001
02	DID:000000	WWN:00000000	00000000	
03	DID:000000	WWN:00000000	00000000	
03	DID:000000	WWN:00000000	00000000	
04	DID:000000	WWN:00000000	00000000	
05	DID:000000	WWN:00000000	00000000	
06	DID:000000	WWN:00000000	00000000	
07	DID:000000	WWN:00000000	00000000	
08	DID:000000	WWN:00000000	00000000	

Figure 64: Example of Device Rows with a Selected LUN to Boot with WWNN

## Canceling a Boot Device

To cancel a boot device:

1. Access the EFI utility (page 57), select **1**, Setup Utility and press <Enter>. The setup utility menu is displayed (Figure 42 on page 59).
2. Select **3**, Configure Boot Parameters and press <Enter>. The boot parameter configuration menu is displayed (Figure 52 on page 62).
3. Select **4**, Configure Boot Devices and press <Enter>. Device rows with selected LUNs to boot with WWNN or D\_ID are displayed (Figure 64).
4. Select a device row and press <Enter>. The EFI utility menu is displayed (Figure 60).
5. Select **2**, Clear Flash Boot Info and press <Enter>. The system clears the target as the remote boot device. A listing of devices is displayed.

## Changing Delay Device Discovery

This parameter can be used to set a delay to occur after an LIP and before a scan is initiated to discover the target. The delay device discovery parameter defaults to off or 0 seconds.

The delay device discovery parameter should be set to 20 seconds for an HP MSA1000 or HP MSA1500 RAID array. The default of 0 should be used for all other arrays.

1. Access the EFI utility (page 57), select **1**, Setup Utility and press <Enter>. The setup utility menu is displayed (Figure 42 on page 59).
2. Select **3**, Configure Boot Parameters and press <Enter>. The boot parameter configuration menu is displayed (Figure 52 on page 62).
3. Select **5**, Delay Device Discovery. The following window is displayed:

Current Delay Device Discovery is: 0000 seconds  
Delay Device Discovery: 0000

4. Use the left and right arrows to select values in 10 second increments. Press <Enter>.

## Resetting Parameter Defaults for an HBA

Parameter defaults can be reset using the EFI utility. These defaults are listed in Table 6.

**Table 6: HBA Defaults**

Parameter	Default	Valid Values
Enable/Disable BIOS	Disabled	Enabled Disabled
AL_PA Value	0x00 Fibre	See AL_PA reference table
Device Path	Fibre Channel	Fibre Channel (FC) SCSI
PLOGI Retry Timer	Disabled	Disabled 50 msec 100 msec 200 msec
Boot Target Scan	Boot path from NVRAM targets	Boot path from NVRAM targets Boot path discovered targets Do not create boot path
Max LUNs Setting	0256	0 – 4096

Parameter	Default	Valid Values
Topology	Auto (start FC-AL)	Auto (start FC-AL) Point-to-Point Auto (start Point-to-Point) FC-AL
Delay Device Discovery	0000	0000 - 0255

To reset parameters:

1. Access the EFI utility (page 57), select **1**, Setup Utility and press <Enter>. The Setup utility menu is displayed (Figure 42 on page 59).
2. Select **4**, Reset to Default Values. The following message is displayed:

Do you want to force defaults (Y/N)?

Figure 65: Force Defaults Dialog

3. Type **Y**. The HBA reverts back to all factory settings (defaults).

---

**Note:** Press <Esc> to return to the EFI utility menu.

---

## Installing the Server

### Install Windows Server 2003 (Itanium 64-bit) on a Fibre Channel Disk

This procedure installs the Windows Server 2003 (Itanium 64-bit) operating system onto a previously unformatted Fibre Channel disk drive that is connected to the Emulex adapter.

Prerequisites:

- This procedure requires that your system have a floppy disk drive. If your system does not have a floppy disk drive, refer to the server setup CD that was shipped with your system.

To install Windows Server 2003 on an Itanium 64-bit platform:

1. From the Emulex Web site or the CD-ROM that accompanies your Emulex product, download the latest version of the Emulex driver for Windows Server 2003 (Itanium 64-bit). Download the FC Port driver, SCSIport Miniport driver or Storport Miniport driver depending on your requirements.
2. Extract the driver files onto a formatted floppy disk.
3. Boot with the Windows Server 2003 Setup Media. This procedure assumes you are using CD-ROM medium. See Microsoft documentation for details on Windows Server 2003.
4. The following message is displayed in the status bar:

Press F6 if you need to install a third party SCSI or RAID driver.  
Press <F6> within five seconds.

5. Press <F6> within five seconds to load the Emulex driver. Watch for the following message:

Setup could not determine the type of one or more mass storage devices installed in your system, or you have chosen to manually specify an adapter.

6. Insert the Emulex driver disk that you created in step 2. Press <Enter>.
7. Follow the instructions to complete the Emulex installation.

8. Remove the floppy disk that contains the driver files.
9. Complete the Windows Server 2003 installation.

## Booting from SAN with Linux and Emulex HBAs

See *Tools for the Emulex Driver for Linux* on the Emulex Web site. (Navigate to the Support area, click the link for Emulex software and documentation, click the Linux link, and click the Linux Tools graphic.)

## EFIBoot Reference

---

### Driver Handle Table Column Definitions

To view Emulex driver handle information, at the shell prompt type:

```
fs0:> drivers
```

A list of drivers is displayed. Figure 40 on page 58 is an example of driver handle information.

### Driver Handle Table Column Definitions

Drv - the device driver handle number.

Version - the driver version.

Type - the device type. D indicates a device driver. B indicates a bus driver.

Cfg - if this column is marked with an X, the driver is supported by the EFI utility (configuration protocol).

Diag - if this column is marked with an X, the driver is supported by the EFI diagnostic protocol.

#D - the number of devices for the driver.

#C - the number of child devices for the driver (bus drivers only).

Driver Name and Image Name - self explanatory.

## Topology Information

### Arbitrated Loop Operation

- FC-AL (Loop) topology only is used. After successful loop initialization, the driver attempts login with FL\_PORT (switched fabric loop port).
- If FL\_PORT login is successful, public loop operation is entered.
- If FL\_PORT login is unsuccessful, private loop mode is entered.

### Point-to-Point Operation

- If F\_PORT (point-to-point) login is successful, fabric mode is used.
- If F\_PORT login is unsuccessful, N\_PORT-to-N\_PORT direct connection topology is used.
- If a switch is discovered, the driver performs the following tasks:
  - FL\_PORT login (FC-AL topology) or F\_PORT login (Point-to-Point topology).
  - Simple Name Server login.
  - State Change Registration.
  - Symbolic Name Registration.
  - FCP Type Registration if RegFcpType is set to 1.
  - The driver logs out and re-logs in. The name server indicates that registration is complete.



- Simple Name Server Query for devices (the registry parameter SnsAll determines whether all N\_PORTS are requested or only SCSI FCP N\_PORTS).
- Discovery/device creation occurs for each target device described by the name server.
- RSCN and LOGO/PRLO are handled by the driver. Reception of either causes new discovery/logins to take place.

---

**Note:** In a fabric environment, the order that disk devices are created based on the name server response data (which is not guaranteed to be in any special order). Between successive boots, the same device may be identified with a different physical device number. However, any devices that have been assigned a device letter through the disk administrator continue to use that letter regardless of the physical device number.

---

## Valid AL\_PA Values

**Table 7: Valid AI\_PA Values**

0x00	0x01	0x02	0x04	0x08	0x0F	0x10	0x17
0x18	0x1B	0x1D	0x1E	0x1F	0x23	0x25	0x26
0x27	0x29	0x2A	0x2B	0x2C	0x2D	0x2E	0x31
0x32	0x33	0x34	0x35	0x36	0x39	0x3A	0x3C
0x43	0x45	0x46	0x47	0x49	0x4A	0x4B	0x4C
0x4D	0x4E	0x51	0x52	0x53	0x54	0x55	0x56
0x59	0x5A	0x5C	0x63	0x65	0x66	0x67	0x69
0x6A	0x6B	0x6C	0x6D	0x6E	0x71	0x72	0x73
0x74	0x75	0x76	0x79	0x7A	0x7C	0x80	0x81
0x82	0x84	0x88	0x8F	0x90	0x97	0x98	0x9B
0x9D	0x9E	0x9F	0xA3	0xA5	0xA6	0xA7	0xA9
0xAA	0xAB	0xAC	0xAD	0xAE	0xB1	0xB2	0xB3
0xB4	0xB5	0xB6	0xB9	0xBA	0xBC	0xC3	0xC5
0xC6	0xC7	0xC9	0xCA	0xCB	0xCC	0xCD	0xCE
0xD1	0xD2	0xD3	0xD4	0xD5	0xD6	0xD9	0xDA
0xDC	0xE0	0xE1	0xE2	0xE4	0xE8	0xEF	

## Installing the EFI Utility

Prerequisites:

- The EFIBoot zip file has been downloaded and the contents extracted to a bootable media. The EFIBoot zip file contents include the .prg file, the elxcli.efi file and the readme.txt file. The elxcli.efi file contains the EFI utility.

To install the EFI utility:

1. Start your system. A menu similar to the following is displayed. The menu items vary depending on the configuration of your system.

```
EFI Boot Manager ver 1.10 [14.61] Firmware ver 2.01 [4322]
Please select a boot option
Red Hat Linux Advanced Workstation
Windows XP 64-Bit Edition Version 2003
EFI Shell [Built-in]
CDROM
Boot Option Maintenance Menu
System Configuration Menu

Use ↑ and ↓ to change option(s). Use Enter to select an option
```

Figure 66: Boot Option Menu

2. Select **EFI Shell** from the menu. An EFI shell is started and the shell prompt is displayed.  
Shell>
3. Change to the drive where the Emulex EFIBoot file was extracted. To see a list of drives, type:  
Shell> map -r<Enter>
4. A list of drive names is displayed (Figure 67)

```
Shell> map -r
Device mapping table
fs0 : Acpi(PNP0A03,0)/Pci(1F|0)/Acpi(PNP0604,0)
blk0 : Acpi(PNP0A03,0)/Pci(1F|0)/Acpi(PNP0604,0)
blk1 : Acpi(PNP0A03,0)/Pci(1F|1)/Ata(Secondary,Master)
Shell>
```

Figure 67: Drive List Example

5. Change to the desired drive by entering the drive name followed by a colon on the command line (for example, "fs0"). Load the EFI utility file by typing "load *filename.efi*" and pressing <Enter> at the shell prompt. In the following example, the Emulex EFI utility file, elxcli311a5.efi, is on drive fs0.

```
fs0:> load elxcli311a5.efi
```

The following message is displayed:

```
fs0:> load elxcli311a5.efi
```

```
load:Image fs0: \elxcli311a5.efi loaded at 0x3FCD7000 - Success
```

The EFI utility is installed.

## Unloading EFIBoot

To unload EFIBoot from the EFI shell:

1. Type:  
fs0:> drivers
2. Press <Enter>. A list of drivers is displayed (Figure 40 on page 58).
3. Write down the Emulex driver handle. In Figure 40, the Emulex driver handle number is 86.

4. To unload the Emulex SCSI Pass Thru driver, type:

Unload *<Emulex driver handle>*

where *<Emulex driver handle>* is the driver handle number. The SCSI Pass Thru driver handle number in the example is 86.

5. Press <Enter>. EFIBoot is unloaded.





## Troubleshooting

There are circumstances in which your system may operate in an unexpected manner. The Troubleshooting section explains several of these circumstances and offers one or more workarounds for each situation.

### Troubleshooting x86 BootBIOS

---

#### The Bootup Message Does Not Appear As the System Boots

**Situation:** You want to access the BIOS utility, but the bootup message does not appear.

**Resolution:** Make sure that x86 BootBIOS has been loaded and enabled.

#### Retry This Adapter Message

**Situation:** The message "Retry This Adapter" appears during BIOS scanning.

**Resolution:** Check the hardware configuration or reconfigure the HBA BIOS using the BIOS utility.

#### Cannot Mount Root File System Message (Solaris SFS Driver with MPxIO Enabled)

**Situation:** The message "Cannot Mount Root File System" appears during bootup.

**Resolution:** Make sure the correct storage device is identified in the `scsi_vhci.conf` file. The XP128 storage array is used in the following example:

```
# cd /kernel/drv
# pg scsi_vhci.conf
#
# Copyright 2004 Sun Microsystems, Inc. All rights reserved.
# Use is subject to license terms.
#
# pragma ident    "@(#)scsi_vhci.conf    1.9    04/08/26 SMI"
#
name="scsi_vhci" class="root";
.
.
.
device-type-scsi-options-list =
"HP    OPEN-3*4", "symmetric-option";
symmetric-option = 0x1000000;
#
```

#### Cannot Find UNIX Kernel Message (Solaris SFS Driver)

**Situation:** The message "Cannot Find UNIX Kernel" appears during bootup.

**Resolution:** Set up the correct LUN to boot in the BIOS utility. The correct LUN can be seen at the end of the Device Address line when you issue a `luxadm display <device>` command; see the `luxadm` documentation from Sun for more information.

#### No Such Partition Message (Solaris SFS Driver)

**Situation:** The message "No Such Partition" appears during bootup:

**Resolution:** Make sure the correct boot device is selected at the GRUB menu. See the GRUB documentation from Sun and the `/boot/grub/menu.lst` for more details.

## Troubleshooting OpenBoot

---

### The System Cannot mount or fsck /etc/vfstab a Fibre Channel Boot Disk (Solaris Ipfc Driver)

**Situation:** During the boot process, the system cannot mount or fsck /etc/vfstab a Fibre Channel boot disk.

**Resolution:** Make sure that persistent binding is implemented correctly.

### A Loaded File Is Not Executable (Solaris Ipfc Driver)

**Situation:** After entering boot disk, a message states that the file that was loaded is not executable.

**Resolution:** The boot block may not be installed correctly to the Fibre Channel drive. See *Making a Bootable System Disk (Solaris Ipfc Driver)* on page 50.

### The System Hangs or Reboots After Displaying Driver Information (Solaris Ipfc Driver)

**Situation:** The system hangs for a long time after displaying driver information, or it reboots after displaying driver information.

**Resolution:** Possible incorrect topology set in the /kernel/drv/lpfc.conf file on the target disk.

### Fibre Channel Disk Is Not Found (Solaris Ipfc Driver)

**Situation:** You have performed the setup tasks and the Fibre Channel disk is not found when you reboot the system.

**Resolution:** If the Fibre Channel disk is not found when the system is rebooted, it may be necessary to do the following:

1. Type "cfgadm -a" to list the target.
2. Type "cfgadm -vc configure c1::c5t2200002037AE0091" to configure the FC target.
3. Type "cfgadm -c unconfigure c1" to remove the FC target.

It may also be necessary to add an entry for the boot drive to the sd.conf file.

## Troubleshooting EFIBoot

---

The Driver Diagnostics Protocol is run from the EFI shell. This protocol allows you to run standard diagnostics as follows:

- For all devices in your system
- For all devices managed by a specific driver
- For all child devices of a specific device, managed by a specific driver
- For a specific child device of a specific device, managed by a specific driver

### Diagnostics Overview

To view diagnostic functions:

1. Type:  
    drvdiag

2. Press < Enter >. The following protocol is displayed:

```
DRVDIAG [-c] [-lXXX] [-s] [-e] [-m] [DriverHandle [DeviceHandle
[ChildHandle] ] ]
```

**Table 8: Diagnostic Protocol**

Protocol	Description
-c	Diagnose all child devices
-lXXX	Diagnose using the ISO 639-2 language specified by XXX
-s	Run diagnostics in standard mode
-e	Run diagnostics in enhanced mode (internal loopback test)
-m	Run diagnostics in manufacturing mode (external loopback test)
DriverHandle	The handle of the driver to configure
DeviceHandle	The handle of a device that DriverHandle is managing
ChildHandle	The handle of a device that is a child of DeviceHandle

## Showing Available Drivers and Controllers

To show driver information:

1. To view Emulex driver handle information, at the shell prompt type:

```
fs0:> drivers
```

Driver information is displayed (Figure 40 on page 58).

---

**Note:** To view driver handle information one page at a time, type:

```
fs0:> drivers -b
```

---

2. To view drivers and controllers for which you can run diagnostics, enter the following command:

```
drvdiag
```

3. Press <Enter>. Information similar to Table 9 is displayed. In this example, 86 is the driver handle and 41 through 48 are the controller handles. Lang [eng] represents English as the language being used.

**Table 9: Available Drivers and Controllers List Example**

Drv[12]	Ctrl[17]	Lang[eng]
Drv[12]	Ctrl[18]	Lang[eng]
Drv[12]	Ctrl[19]	Lang[eng]
Drv[12]	Ctrl[20]	Lang[eng]
Drv[12]	Ctrl[21]	Lang[eng]
Drv[12]	Ctrl[22]	Lang[eng]
Drv[12]	Ctrl[23]	Lang[eng]
Drv[12]	Ctrl[24]	Lang[eng]
Drv[27]	Ctrl[10]	Lang[eng]



**Table 9: Available Drivers and Controllers List Example (Continued)**

Drv[2C]	Ctrl[33]	Lang[eng]
Drv[2C]	Ctrl[34]	Lang[eng]
Drv[2D]	Ctrl[8]	Lang[eng]
Drv[2D]	Ctrl[9]	Lang[eng]
Drv[2E]	Ctrl[11]	Lang[eng]
Drv[2F]	Ctrl[14]	Lang[eng]
Drv[35]	Ctrl[04]	Lang[eng]
Drv[36]	Ctrl[19]	Lang[eng]
Drv[37]	Ctrl[06]	Lang[eng]
Drv[38]	Ctrl[23]	Lang[eng]
Drv[39]	Ctrl[17]	Lang[eng]
Drv[39]	Ctrl[18]	Lang[eng]
Drv[3A]	Ctrl[28]	Lang[eng]
Drv[3A]	Ctrl[29]	Lang[eng]
Drv[3A]	Ctrl[29]	Lang[eng]
Drv[3B]	Ctrl[13]	Lang[eng]
Drv[3B]	Ctrl[24]	Lang[eng]
Drv[3C]	Ctrl[19]	Lang[eng]
Drv[3D]	Ctrl[02]	Lang[eng]
Drv[3D]	Ctrl[03]	Lang[eng]
Drv[3E]	Ctrl[39]	Lang[eng]
Drv[3E]	Ctrl[40]	Lang[eng]
Drv[4B]	Ctrl[27]	Lang[eng]
Drv[4B]	Ctrl[28]	Lang[eng]
Drv[4B]	Ctrl[29]	Lang[eng]
Drv[4C]	Ctrl[11]	Lang[eng]
Drv[4C]	Ctrl[12]	Lang[eng]
Drv[4D]	Ctrl[06]	Lang[eng]
Drv[4E]	Ctrl[35]	Lang[eng]
Drv[4E]	Ctrl[36]	Lang[eng]
Drv[4E]	Ctrl[37]	Lang[eng]
Drv[4E]	Ctrl[38]	Lang[eng]
Drv[51]	Ctrl[17]	Lang[eng]
Drv[52]	Ctrl[42]	Lang[eng]
Drv[53]	Ctrl[07]	Lang[eng]
Drv[54]	Ctrl[18]	Lang[eng]
Drv[86]	Ctrl[41]	Lang[eng]

**Table 9: Available Drivers and Controllers List Example (Continued)**

Drv[86]	Ctrl[42]	Lang[eng]
Drv[86]	Ctrl[43]	Lang[eng]
Drv[86]	Ctrl[44]	Lang[eng]
Drv[86]	Ctrl[45]	Lang[eng]
Drv[86]	Ctrl[46]	Lang[eng]

## Running Diagnostics

### Running Standard Diagnostics on a Driver

To run standard diagnostic on a driver:

1. To view Emulex driver handle information, at the shell prompt type:  

```
fs0:> drivers
```
2. Determine the driver number, then type:  

```
drvdiag -s <driver number>
```
3. Press <Enter>. Figure 68 is an example of default standard diagnostic information for driver 22.

```
fs0:\> drvdiag -s 22
Run Diagnostics
Drv[22] Ctrl[23] Lang[eng]
Emulex Efi Driver Diagnostic - Standard:
-----
Controller Location : PCI Bus#: 60 Device#: 01 Function#: 00
Controller Name : LP1050DC PCI-X 2Gb/s 133MHz
Power on self Test : Passed
Controller Initialize : Success
Emulex Firmware Mode : SLI-1/2
Controller Boot Status : Boot Bios Disabled
Run Emulex EFI Setup Utility to Enable Adapter
- PASSED
```

*Figure 68: Default Standard Diagnostic Information Example*

### Running Standard Diagnostics on a Child Device

Prerequisites:

- The BIOS must be enabled.
- LUNs must be mapped.

To run standard diagnostics on a child device:

1. To view Emulex driver handle information, at the shell prompt type:  

```
fs0:> drivers
```
2. Determine the driver number, device number and child number, then type:  

```
drvdiag -s <driver number> <device number> <child number>
```

3. Press <Enter>. Figure 69 is an example of default standard information for driver 22, device 23 and child 88.

```

fs0:\> drvdiag -s 22 23 88
Run Diagnostics
Drv[22] Ctrl[23] Child[88] Lang[eng]
Emulex Efi Driver Diagnostic - Standard:
-----
Controller Location : PCI Bus#: 60 Device#: 01 Function#: 00
Controller Name : LP1050DC PCI-X 2Gb/s 133MHz
Power on self Test : Passed
Controller Initialize : Success
Emulex Firmware Mode : SLI-1/2
Controller Boot Status : OK
PCI Loopback Test : Passed
Fibre Channel Link : Up
Device Discovery : Done
Number of Target(s) : 1

Target(s) List :
Index RPI PortId WorldWideNodeNumber
000 001 0000C9 50020482BFD04A54

Device Name : SUPER RAID 4468
Device Path : Fibre
World Wide Node Number : 50020482BFD04A54
Logical Unit Number : 0000000000000000
Device Block Size : 512 Bytes.
Number of Blocks (LBA) : 00778800
Device Capacity : 0003 Gigabytes.
Read LBA (0)
0000 33 C0 8E D0 BC 00 7C FB 50 07 50 1F FC BE 1B 7C 3 P P
.
.
.
1F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
- PASSED

```

Figure 69: Default Standard Diagnostic Information for a Child Device

---

**Note:** Multiple lines of output are displayed for the logical block address (LBA).  
 To display diagnostics for all child devices for driver 22, device 23, type:  
 drvdiag -s 22 23 -c

---

### Running Enhanced Diagnostics on a Device

The enhanced diagnostic performs an internal loopback test. This diagnostic tests the PCI bus to the HBA. Some commands are echoed and then returned to the host.

To run enhanced diagnostics on a device:

1. To view Emulex driver handle information, at the shell prompt type:  
 fs0:> drivers
2. Determine the driver number and device number, then type:  
 drvdiag -e <driver number> <device number>

3. Press <Enter>. Figure 70 is an example of the output for driver 22 and controller handles 23 and 24.

```
fs0:\> drvdiag -e 22
Run Diagnostics
Drv[22] Ctrl[23] Lang[eng] - PASSED
Drv[22] Ctrl[24] Lang[eng] - PASSED
```

Figure 70: Enhanced Diagnostic Information for a Device

## Running Manufacturing Diagnostics on a Device

The manufacturing diagnostic performs an external loopback test. This diagnostic tests the HBA to the SAN.

Prerequisites:

- A loopback connector is in place for each Fibre Channel port in the storage area network (SAN).

To run manufacturing diagnostics on a device:

1. To view Emulex driver handle information, at the shell prompt type:

```
fs0:> drivers
```

2. Determine the driver number and device number, then type:

```
drvdiag -m <driver number> <device number>
```

3. Press <Enter>. Figure 71 is an example of the output for driver 22 and controller handles 23 and 24.

```
fs0:\> drvdiag -e 22
Run Diagnostics
Drv[22] Ctrl[23] Lang[eng] - PASSED
Drv[22] Ctrl[24] Lang[eng] - PASSED
```

Figure 71: Manufacturing Diagnostic Information for a Device