Emulex® Drivers for Linux

User Guide
Release 12.6

Broadcom reserves the right to make changes without further notice to any products or data herein to improve reliability, function, or design. Information furnished by Broadcom is believed to be accurate and reliable. However, Broadcom does not assume any liability arising out of the application or use of this information, nor the application or use of any product or circuit described herein, neither does it convey any license under its patent rights nor the rights of others.

Table of Contents

Chapter 1: Introduction

1.1 Overview

This guide provides installation, updating, uninstalling, configuring, and troubleshooting procedures for several types of drivers for Linux that are supported by Emulex®.

This guide is applicable to several versions of drivers, operating systems, firmware, and adapters. For supported firmware versions and their latest release, go to www.broadcom.com.

This product supports the following Emulex FC HBAs:

- LPe12000-series adapters
- LPe16000-series adapters
- LPe31000-series adapters
- LPe32000-series adapters
- LPe35000-series adapters

For a list of Emulex adapters that are compatible with the stand-alone driver kits go to www.broadcom.com.

1.2 Abbreviations

The following table lists the acronyms and abbreviations used in this document.

Table 1: Acronyms and Abbreviations

Acronym/Abbreviation	Description
ADISC	Address discovery
ANA	asymmetric namespace access
CDB	Command Descriptor Block
CQ	completion queue
CQID	completion queue identifier
CT	common transport
DIF	data integrity field
DIMM	dual inline memory module
DIX	Data Integrity Extension
DM	Device Mapper
DMA	direct memory access
ELS	Extended Link Service
EQ	Event Queue
EQE	event queue entry
FAN	File Area Network
FCFI	Fibre Channel Forwarder Indicator
FCP	Fibre Channel Protocol
FDMI	Fabric-Device Management Interface
FLOGI	fabric login
GnuPG	GNU Privacy Guard

Table 1: Acronyms and Abbreviations (Continued)

Acronym/Abbreviation	Description
I/O	input/output
IOCB	input/output command block
IOCBQ	input/output command block queue
IOPs	input/output operations per second
IOTAG	input/output tag
IP CSUM	IP checksum
ISR	interrupt service routine
LPFC	Light Pulse Fibre Channel
MB	megabyte
MDS	Multilayer Director Switch
MQ	MultiQueue
MSI	message signaled interrupt
MSI-X	extended message signaled interrupt
NAA	Network Address Authority
NDLP	Network Data Loss Prevention
NPIV	N_Port ID Virtualization
PLOGI	Port Login
RPI	remote port indicator
RPM	resource package manager
RQ	Receive Queue
RSCN	registered state change notification
Rx	receive
SGE	Oracle (formerly Sun) grid engine
SGL	scatter-gather list
ULP	upper layer protocol
VPI	NPIV port ID
WQ	work queue

Chapter 2: Installing and Uninstalling

The Linux driver supports both FCP and NVMe over FC.

Emulex releases Linux binary RPMs that are digitally signed using the GNU Privacy Guard (GnuPG) standard. This allows certification of the contents of the RPMs and verification that the contents have not been modified since they were created by Emulex. The RPMs have been digitally signed by Emulex with a GnuPG private key that is held only by Emulex. Instructions for creating the GnuPG public key file are located at:

https://docs.broadcom.com/docs/1334566

2.1 General Installation Requirements

NOTE: Update the latest driver on the operating system before updating the firmware.

Prior to driver installation, follow these general requirements:

- Install a supported Emulex adapter in the system. Refer to the adapter's installation guide for specific hardware installation instructions.
- Use a supported operating system. Refer to www.broadcom.com for supported operating systems.
- Install the latest driver.
- Update to the latest firmware on the adapter.

NOTE: For LPe35000-series adapters only:

In some cases, a firmware update requires either a firmware reset or a basic PCIe reset, depending on the features available in the new firmware. A firmware reset is performed automatically if it is needed, regardless of the operating system. A basic PCIe reset is also performed automatically, but only on the following Linux operating systems:

- SLES 12 SP4 and later
- SLES 15 and later
- RHEL 7.6 and later
- RHEL 8.0 and later

In some cases, a full reboot is required to activate new firmware or to enable a new feature. In that case, a message similar to one of the following appears after the firmware download is complete:

```
Download successfully completed. Please reboot the system to activate new firmware.
```

```
Download completed. Some features require an optional reboot. Refer to the Adapter's
Firmware and Boot Code Release Notes for details.
```

For a list of features that require a reboot to be enabled, refer to the *Emulex LPe35000-Series HBA Firmware and Boot Code Release Notes*.

2.2 Emulex FC Driver Kit

The Emulex FC driver kit contains the following:

- A zipped tar file that includes the driver binary RPMs for a specific driver version and Linux distribution. On select driver kits, the zipped tar file may also include any of the following RPMs:
  - `nvmefc-connect` RPM
  - `elx-lpfc-vector-map` RPM
  - `elx-lpfc-extras` RPM

  NOTE: Use only officially released Linux distribution kernels. The binary RPM packages support only officially released Linux distribution kernels and do not support prerelease distribution kernels. Refer to the *Emulex Drivers for Linux Release Notes* for details of kernel versions that are supported by the driver.
- An installation script, `elx_lpfc_install.sh`, that installs the FC driver binary RPM that corresponds to the target system's architecture and kernel memory variant.
- A `README` file that provides a description of the kit structure, its contents, and distribution support scope.

2.2.1 Installing the Emulex FC Driver Kit

NOTE: Installing a newer version of the FC driver kit automatically removes any previous versions of the FC driver kit.

After the operating system is installed, perform the following steps to install the Emulex FC driver:

1. Download driver kit version 12.4 or later from www.broadcom.com for the following operating systems:
   - RHEL 8.0 and later
   - RHEL 7.5 and later
   - SLES 15 and later
   - SLES 12 SP4 and later
2. Log in to a terminal as `root`.
3. Untar the driver kit by typing the following command:
   ```
   tar -zxvf elx-lpfc-dd-<Linux_distribution_version>-<driver_version>-ds-n.tar.gz
   ```
4. Change to the directory into which the driver kit was extracted by typing the following command:
   ```
   cd elx-lpfc-dd-<Linux_distribution_version>-<driver_version>-ds/
   ```
   - For installing only the Emulex FC driver kit:

     Run the `elx_lpfc_install.sh` script without options to install the driver kit:
     ```
     ./elx_lpfc_install.sh
     ```
     After the `elx_lpfc_install.sh` script has finished running successfully, the Emulex FC driver is loaded, and devices that are connected to the system are accessible.
   - For installing the Emulex FC driver kit with NVMe:

     NOTE: For instructions on setting up boot from SAN on NVMe over FC in Linux, refer to the *Emulex Boot for the Fibre Channel Protocol User Guide*.

     Ensure that `nvme-cli` is installed; otherwise the script `./elx_lpfc_install.sh -n` will fail.

     To confirm if `nvme-cli` is installed, run the `rpm -qa | grep nvme-cli` command.

     Run the `elx_lpfc_install.sh` script with the `-n` option:
     ```
     ./elx_lpfc_install.sh -n
     ```
     This installs the `lpfc` driver along with `nvmefc-connect rpm`.

     After the `elx_lpfc_install.sh` script has finished running successfully, the NVMe over FC driver is installed.

5. Reboot the system to enable the newly added driver options in the ramdisk.
6. You can use the following command to verify if the version of the newly installed LPFC driver is correct:

```
cat /sys/module/lpfc/version
```

2.2.2 Uninstalling the Emulex FC Driver Kit

NOTE: You must run the uninstall script that shipped with the version of the driver kit that you want to remove.

To uninstall the Emulex FC driver, perform the following steps:

1. Log in as `root`.
2. If possible, exit all applications that use FC-attached drives, and then unmount the drives. If you cannot exit all applications that use FC-attached drives, the uninstall script works properly, but you must reboot after the uninstall is complete.
3. Run the `elx_lpfc_install.sh` script with the `--uninstall` option:

```
./elx_lpfc_install.sh --uninstall
```

NOTE: The `--uninstall` options also uninstalls `elx-lpfc-vector-map`, `elx-lpfc-extras`, and `nvmefc-connect`.

2.3 Booting from a Nonzero LUN Attached to an Emulex FC Adapter

To configure SLES 12 SP*x* or SLES 15 SP*x* to boot from an FC-attached disk device other than a LUN 0 device, refer to the *Boot for the Fibre Channel Protocol User Guide,* which is available at www.broadcom.com.

2.4 Emulex HBA Manager Application

NOTE: The Emulex OneCommand® Manager application has been renamed as the Emulex HBA Manager application.

The Emulex HBA Manager application is a powerful and centralized adapter management suite. It provides discovery, reporting, and management of local and remote adapters from a single console anywhere in the SAN and across platforms. Both a GUI and a CLI are provided. For instructions on installing and using the Emulex HBA Manager application, refer to the *Emulex HBA Manager User Guide*, which is available at www.broadcom.com.

2.5 Updating the Firmware

You can update the firmware on the adapter using any of the following options:

- Using the Emulex HBA Manager. For information on how to update the firmware using the Emulex HBA Manager, refer to the *Emulex HBA Manager User Guide*.
- Using HBACMD. For information on how to update the firmware using HBACMD, refer to the *Emulex HBA Manager Command Line Interface User Guide*.
- Using `sysfs`.

Updating the Firmware Using sysfs:

To update the firmware using `sysfs`, perform the following steps:

1. Before you update the firmware, procure the model name of the adapter.
   ```
   cat /sys/class/scsi_host/<host_no>/modelname
   ```
2. Modify the firmware file name to the model name.
3. Copy the firmware file to `/lib/firmware` with the model name.
   ```
   cp <firmware_file>.grp /lib/firmware/
   ```
4. Run the following command and note the value for `lpfc_log_verbose`. You must echo the same value to `lpfc_log_verbose` after firmware activation.
   ```
   cat /sys/class/scsi_host/host<X>/lpfc_log_verbose
   ```
5. Set the driver parameter `lpfc_log_verbose=0x8`.
   ```
   echo 0x8 > /sys/class/scsi_host/host<X>/lpfc_log_verbose
   ```
6. Download the firmware to the adapter.
   ```
   echo 1 > /sys/class/scsi_host/<host_no>/lpfc_req_fw_upgrade
   ```
7. Check the `dmesg` log to determine the appropriate reset action.
8. Depending on the features available in the new firmware, the firmware update requires either a firmware reset or a PCI bus reset.

   To perform a firmware reset, issue the following command:
   ```
   echo fw_reset > /sys/class/scsi_host/host<X>/board_mode
   ```
   To perform a PCI bus reset, perform the following steps:

   a. Run the following command and note the value for `lpfc_enable_hba_reset`. You must echo the same value to `lpfc_enable_hba_reset` after firmware activation.
   ```
   cat /sys/class/scsi_host/host<X>/lpfc_enable_hba_reset
   ```
   b. Enable HBA reset on the adapter.
   ```
   echo 2 > /sys/class/scsi_host/host<X>/lpfc_enable_hba_reset
   ```
   c. Set *all ports* of the adapter offline.
   ```
   echo offline > /sys/class/scsi_host/host<X>/board_mode
   ```
   d. Reset the PCI bus of the adapter.
   ```
   echo pci_bus_reset > /sys/class/scsi_host/host<X>/board_mode
   ```
   e. Set *all ports* of the adapter online.
   ```
   echo online > /sys/class/scsi_host/host<X>/board_mode
   ```
9. Enter the value of `lpfc_enable_hba_reset` procured in Step a.
   ```
   echo <original_value> /sys/class/scsi_host/host<X>/lpfc_enable_hba_reset
   ```
10. Enter the value of `lpfc_log_verbose` procured in Step 4.
    ```
    echo <original_value> /sys/class/scsi_host/host<X>/lpfc_log_verbose
    ```

2.6 Installing the NVMe over FC Driver

This section describes how to install the out-of-box NVMe drivers. NVMe drivers are supported on the following operating systems.

- RHEL 8.0 and later
- RHEL 7.6 and later
- SLES 15 and later
- SLES 12 SP4 and later

NOTE: The NVMe over FC target drivers are meant for evaluation and testing during development only. They are not intended for production use.

NVMe over FC is supported on LPe31000-series, LPe32000-series, and LPe35000-series adapters.

2.6.1 Installing the NVMe over FC Driver on SLES Operating Systems

Install the SLES operating system on the computer, following the instructions provided with the operating system. Installing the operating system automatically installs an inbox initiator and target NVMe driver.

Installation also allows you to install the `nvme-cli` utility and the `nvmetcli` utility. The `nvme-cli` utility, which is required on initiators and on targets that use NVMe disks, supports initiation of new NVMe over fabric connections, and it functions as a general utility that allows you to query and manipulate an NVMe device. The `nvmetcli` utility, which is required on targets only, supports the configuration of a system to be an NVMe over fabric target device.

At this time, Emulex supports NVMe over FC on target devices on SLES 12 SP4 and SLES 12 SP5. During the installation, configure the following items:

- Select the system role of **Default System**.
- In **Installation Settings**, select the following features:
  - Install **NVMe CLI**.
  - Install **NVMeT CLI**.

NOTE: For NVMe over FC target systems, you must update to the latest SUSE kernel and the latest SUSE release of the NVMe CLI and NVMeT CLI packages. Refer to the *Emulex Drivers for Linux Release Notes* for more information.

2.6.2 Installing the NVMe over FC Driver on RHEL Operating Systems

Install the RHEL operating system on the computer, following the instructions provided with the operating system. Installing the operating system automatically installs an inbox initiator NVMe driver. At this time, Emulex supports NVMe over FC on RHEL 7.6 and later and RHEL 8.0 and later.

During the installation, select the system role of **Default System**.

After the operating system is installed, extract the `nvme-cli` utility from the operating system .ISO image. The `nvme-cli` utility supports initiation of new NVMe over fabric connections, and it functions as a general utility that allows you to query and manipulate an NVMe device. To install the `nvme-cli` utility, type this command:

```
rpm -ivh nvme-cli-x.x.x.rpm
```

You must update to the latest RHEL kernel and the latest RHEL release of the NVMe CLI package. Refer to the *Emulex Drivers for Linux Release Notes* for more information.

Chapter 3: Configuration

3.1 FC Driver Configuration

This section describes how to configure parameters for the FC driver.

3.1.1 FC Driver Parameters

The FC driver parameters determine some aspects of the driver's behavior. There are two main types, static and dynamic. Changes to the static parameters require a driver reload for the changes to take effect. Changes to most dynamic parameters take effect immediately; some do not take effect until there is a link-down and link-up sequence.

For more information on driver parameters, see Section 3.1.1.1, Static FC Driver Parameters, and Section 3.1.1.2, Dynamic FC Driver Parameters.

3.1.1.1 Static FC Driver Parameters

Changes to static parameters require a driver reload for the changes to take effect. The following table lists the static FC driver parameters.

Table 2: Static FC Driver Parameters

Parameter	Description	sysfs Visible	Applicable Adapters	Activation
lpfc_ack0	For Emulex LPe12000-series adapters, when enabled, ACK0 is used for Class 2. The enabled value is 1. The disabled value is 0 (default).	Yes	LPe12000-series adapters	Driver reload
lpfc_discovery_threads	Specifies the maximum number of ELS commands that can be outstanding for a discovery. **NOTE:** The lpfc_discovery_threads parameter defaults to a value of 64 for private loop topologies regardless of the configured value. If multiple ports are configured on the host, the value of 64 is used only for those ports that are connected in a private loop topology. The configured value is used for all other ports. The minimum value is 1. The maximum value is 64. The default value is 32.	No	All	Driver reload
lpfc_enable_bg	Enables the BlockGuard (T10-DIF) feature. The minimum value is 0 (default). The maximum value is 1.	Yes	All	Driver reload
lpfc_enable_da_id	When enabled, the FC driver issues a DA_ID CT command to the fabric when vPorts log out of the fabric. The enabled value is 1 (default). The disabled value is 0.	Yes	All	Driver reload
lpfc_EnableXLane	For Emulex LPe16000-series, LPe31000-series, LPe32000-series, and LPe35000-series adapters this parameter specifies whether the ExpressLane™ feature is enabled or disabled. The minimum value is 0 (default). The maximum value is 1.	Yes	LPe16000-series, LPe31000-series, LPe32000-series, and LPe35000-series adapters	Driver reload

Table 2: Static FC Driver Parameters (Continued)

Parameter	Description	sysfs Visible	Applicable Adapters	Activation
lpfc_enable_fc4_type	For LPe31000-series, LPe32000-series, and LPe35000-series adapters, when enabled, this parameter defines the FC4 types that are supported. The possible values are: ■ 1 = Enables just FCP. ■ 3 = Enables both FCP and NVMe (default). The default value is 3.	Yes	LPe31000-series, LPe32000-series, and LPe35000-series adapters	Driver reload
lpfc_enable_hba_heartbeat	When enabled, the heartbeat logic in the FC driver is able to detect whether the adapter is functional. If the heartbeat logic detects that the adapter is not functional, the driver will shut down the adapter. The disabled value is 0 (default). The enabled value is 1.	Yes	All	Driver reload
lpfc_enable_hba_reset	When enabled, the FC driver can pass resets to the adapter. This action is typically used for debugging purposes. The enabled value is 1 (default). The disabled value is 0.	Yes	All	Driver reload
lpfc_enable_npiv	When enabled, the FC driver can use NPIV to create vPorts (if supported by the fabric). The enabled value is 1 (default). The disabled value is 0.	Yes	All	Driver reload
lpfc_enable_nvmet	For LPe31000-series, LPe32000-series, and LPe35000-series adapters, specifies the WWPN of the ports enabled for NVMe target.	No	LPe31000-series, LPe32000-series, and LPe35000-series adapters	Driver reload
lpfc_enable_SmartSAN	Sets up FDMI support for SmartSAN. The disabled value is 0 (default). The enabled value is 1.	Yes	All	Driver reload
lpfc_fcp_class	Specifies either FC Class 2 or FC Class 3 for FCP data transmission. For Class 2, the value is 2. For Class 3, the value is 3 (default).	Yes	All	Driver reload
lpfc_fdmi_on	Specifies if FDMI support is enabled or disabled. The enabled value is 1 (default). The disabled value is 0.	Yes	All	Driver reload
lpfc_hba_queue_depth	Limits the number of outstanding commands per HBA. The minimum value is 32. The maximum value is 8192 (default).	Yes	All	Driver reload
lpfc_hdw_queue	For LPe16000-series, LPe31000-series, LPe32000-series, and LPe35000-series adapters, defines the number of I/O channels supported by the driver. The default value is 0. A value of 0 means that the driver looks at the number of online CPUs in the system. Use the default value. The supported range is 0 to 128. **NOTE:** This parameter applies to the following operating systems: ■ RHEL 8.0 and later ■ RHEL 7.6 and later ■ SLES 15 and later ■ SLES 12 and later	Yes	LPe16000-series, LPe31000-series, LPe32000-series, and LPe35000-series adapters	Driver reload

Table 2: Static FC Driver Parameters (Continued)

Parameter	Description	`sysfs` Visible	Applicable Adapters	Activation
`lpfc_irq_chann`	For LPe16000-series, LPe31000-series, LPe32000-series, and LPe35000-series adapters, defines the number of EQs (and MSI-X vectors) requested. The default is operating system specific. 0 = "per CPU", limited by the lesser of the number of CPUs and `lpfc_hdw_queue`. **NOTE:** This parameter applies to the following operating systems: ■ RHEL 8.0 and later ■ RHEL 7.6 and later ■ SLES 15 and later ■ SLES 12 and later	Yes	LPe16000-series, LPe31000-series, LPe32000-series, and LPe35000-series adapters	Driver reload
`lpfc_lun_queue_depth`	Specifies the default maximum number of commands sent to a single logical unit (disk drive). The minimum value is 1. The maximum value is 512. The default value is 30. The value of `lpfc_lun_queue_depth` is dependent on the `max_xri` value received. The `max_xri` value might vary per adapter type. **NOTE:** This parameter can be dynamically changed using the Emulex HBA Manager application.	Yes	All	Driver reload
`lpfc_max_luns`	Specifies the highest available LUN ID that is valid, per target. For example, a value of 19 means that LUN IDs from 0 to 19 are valid for the target. The SCSI layer scans each target until it reaches this specified LUN ID. The minimum value is 0. The maximum value is 65535. The default value is 255.	Yes	All	Driver reload
`lpfc_max_scsicmpl_time`	Uses command completion time to control queue depth. The units are in milliseconds. The minimum value is 0 (default). The maximum value is 6000.	Yes	All	Driver reload
`lpfc_nvmet_mrq`	For LPe31000-series, LPe32000-series, and LPe35000-series adapters, configures the allocation of multi-receive queues (NVMe target mode only). The default value is 0 (the driver sets the appropriate value). The maximum value is 16.	Yes	LPe31000-series, LPe32000-series, and LPe35000-series adapters	Driver reload
`lpfc_ras_fwlog_func`	For LPe31000-series, LPe32000-series, and LPe35000-series adapters, configures the function on which firmware logging support will be enabled. The minimum value is 0 (default). The maximum value is 7. Default is logging enabled on Function 0.	Yes	LPe31000-series, LPe32000-series, and LPe35000-series adapters	Driver reload
`lpfc_restrict_login`	When enabled, restricts the vPorts login to remote initiators. The enabled value is 1 (default). The disabled value is 0.	No	All	Driver reload
`lpfc_scan_down`	For FC adapters, when enabled, selects the *scan down* method (scanning the AL_PA from high to low) to assign a SCSI ID. The enabled value is 1 (default). The disabled value is 0.	Yes	FC adapters only	Driver reload

Table 2: Static FC Driver Parameters (Continued)

Parameter	Description	sysfs Visible	Applicable Adapters	Activation
lpfc_sg_seg_cnt	Controls the scatter/gather maximum segment count passed to the FC driver. This variable is applicable per SCSI command. For Emulex LPe12000-series adapters, the minimum value is 64 (default), and the maximum value is 4096. For all other adapters, the minimum value is 64 (default), and the maximum value is 510.	Yes	All	Driver reload
lpfc_sli_mode	For Emulex LPe12000-series adapters, this parameter allows you to force the SLI mode requested by the adapter driver. The possible values are: ■ 0 = Auto-select (default) ■ 2 = SLI-2 ■ 3 = SLI-3 **NOTE:** For all other adapters, this parameter is not applicable and has no effect.	No	LPe12000-series adapters	Driver reload
lpfc_use_msi	When enabled, this parameter determines whether the driver uses MSI or MSI-X. ■ 0 = MSI disabled; INTx mode is used. ■ 1 = MSI; allows a maximum of 32 interrupts. ■ 2 = MSI-X; allows a maximum of 2048 interrupts (default value). **NOTE:** The default is 2. This value reverts to 1 if the system does not support MSI-X. This value reverts to 0 if the system does not support MSI.	Yes	All	Driver reload
lpfc_XLanePriority	For Emulex LPe16000-series, LPe31000-series, and LPe32000-series adapters, when enabled, this parameter defines the frame priority to be used for the ExpressLane feature. The minimum value is 0 (default). The maximum value is 0x7f.	Yes	LPe16000-series, LPe31000-series, and LPe32000-series adapters	Dynamic

3.1.1.2 Dynamic FC Driver Parameters

Changes to dynamic parameters take effect immediately. All `lpfc` dynamic parameters are read and written using `sysfs`. The following table lists the dynamic FC driver parameters.

Table 3: Dynamic FC Driver Parameters

Parameter	Description	`sysfs` Visible	Applicable Adapters	Activation
`lpfc_cr_count`	For Emulex LPe12000-series adapters, this parameter determines the value for I/O coalescing for `lpfc_cr_count` outstanding commands. The minimum value is 1 (default). The maximum value is 255. **NOTE:** For all other adapters, this parameter is not applicable and has no effect.	Yes	LPe12000-series adapters	Port reset
`lpfc_cr_delay`	For Emulex LPe12000-series adapters, this parameter determines the value for I/O coalescing for `lpfc_cr_delay` (milliseconds) outstanding commands. The minimum value is 0 (default). The maximum value is 63. **NOTE:** For all other adapters, this parameter is not applicable and has no effect.	Yes	LPe12000-series adapters	Port reset
`lpfc_devloss_tmo`	Specifies the number of seconds to hold an I/O error when a device disappears. The minimum value is 1. The maximum value is 255. The default value is 30.	Yes	All	Dynamic
`lpfc_enable_auth`	For LPe16000-series, LPe31000-series, LPe32000-series, and LPe35000-series adapters, enables or disables DH-CHAP authentication. The possible values are: 0 = Authentication is disabled (default). 1 = Authentication is enabled. **NOTE:** DH-CHAP and Dynamic D_Port are mutually exclusive. If both features are enabled simultaneously, warning messages will result. You must either disable DH-CHAP using the `lpfc_enable_auth` parameter or disable Dynamic D_Port using the Emulex HBA Manager application.	Yes	LPe16000-series, LPe31000-series, LPe32000-series, and LPe35000-series adapters	Dynamic
`lpfc_enable_bbcr`	For LPe16000-series, LPe31000-series, LPe32000-series, and LPe35000-series adapters, specifies if the Buffer-to-Buffer Credit Recovery feature is enabled or disabled. For Emulex LPe12000-series adapters, this parameter is not applicable and has no effect. The minimum value is 0. The maximum value is 1 (default).	Yes	LPe16000-series, LPe31000-series, LPe32000-series, and LPe35000-series adapters	Port reset

Table 3: Dynamic FC Driver Parameters (Continued)

Parameter	Description	sysfs Visible	Applicable Adapters	Activation
lpfc_enable_mds_diags	For LPe16000-series, LPe31000-series, LPe32000-series, and LPe35000-series adapters, enables or disables the MDS Diagnostics feature. For Emulex LPe12000-series adapters, this parameter is not applicable and has no effect.The minimum value is 0 (default). The maximum value is 1. **NOTE:** The parameter must be disabled (set to 0) after diagnostics are complete.	Yes	LPe16000-series, LPe31000-series, LPe32000-series, and LPe35000-series adapters	Port reset
lpfc_fcp_imax	For LPe16000-series, LPe31000-series, LPe32000-series, and LPe35000-series adapters, specifies the maximum number of fast-path FCP interrupts per second. ■ For RHEL 7.5 and earlier kernels: The default value is 0. If set to 0 the parameter is OFF. The minimum value is 5000. The maximum value is 5,000,000. For the following operating systems: – RHEL 8.0 and later – RHEL 7.6 and later – SLES 15 and later – SLES 12 and later A value of 0 (default) provides dynamic interrupt coalescing that adapts to provide low latency or high IOPs according to I/O load. **NOTE:** For Emulex LPe12000-series adapters, this parameter is not applicable and has no effect.	Yes	LPe16000-series, LPe31000-series, LPe32000-series, and LPe35000-series adapters	Dynamic
lpfc_fcp_io_sched	For LPe16000-series, LPe31000-series, LPe32000-series, and LPe35000-series adapters, determines which algorithm to use when scheduling an FC I/O to an I/O channel. For more information, see Section 3.3, FC Driver Performance Tuning. ■ For RHEL 7.5 and earlier kernels: The value 0 (default) configures “round-robin scheduling.” A value of 1 configures “schedule by CPU.” ■ For the following operating systems: – RHEL 8.0 and later – RHEL 7.6 and later – SLES 15 and later – SLES 12 and later The value 0 configures “schedule by hardware queue” (rather than round robin). A value of 1 (default) configures “schedule by CPU.” For Emulex LPe12000-series adapters, this parameter is not applicable and has no effect.	Yes	LPe16000-series, LPe31000-series, LPe32000-series, and LPe35000-series adapters	Dynamic
lpfc_first_burst_size	Configures the first burst size on targets that support first burst. The minimum value is 0 (default). The maximum value is 65,536 bytes.	Yes	All	Dynamic

Table 3: Dynamic FC Driver Parameters (Continued)

Parameter	Description	sysfs Visible	Applicable Adapters	Activation
lpfc_link_speed	For FC adapters, specifies the FC link speed. The possible values are: ■ 0 = Auto-select (default) ■ 2 = 2 Gb/s ■ 4 = 4 Gb/s ■ 8 = 8 Gb/s ■ 16 = 16 Gb/s ■ 32 = 32 Gb/s **NOTE:** ■ For LPe32000-series and LPe35000-series adapters running 11.2 firmware or later, this parameter is ignored. ■ Setting this option incorrectly can cause the adapter to fail to initialize. ■ If you configure the link speed in a BIOS utility, the link speed can be overwritten by the operating system according to its own configuration settings. To avoid this issue, configure the link speed in both the operating system driver and the Boot BIOS or UEFI driver.	Yes	FC adapters only	Port reset
lpfc_log_verbose	Specifies the log verbosity level of the messages posted by the driver. When enabled this parameter prints extra log messages useful for debugging purpose. The minimum value is 0x0 (default). The maximum value is 0xFFFFFFFF.	Yes	All	Dynamic
lpfc_req_fw_upgrade	For LPe16000-series, LPe31000-series, LPe32000-series, and LPe35000-series adapters, set this parameter to 1 to update the firmware.	Yes	LPe16000-series, LPe31000-series, LPe32000-series, and LPe35000-series adapters	Dynamic
lpfc_suppress_rsp	For LPe16000-series, LPe31000-series, LPe32000-series, and LPe35000-series adapters, negotiates whether to suppress the RSP from a target during a PLOGI. The minimum value is 0. The maximum value is 1. The default value is 1.	Yes	LPe16000-series, LPe31000-series, LPe32000-series, and LPe35000-series adapters	Dynamic
lpfc_tgt_queue_depth	Limits the number of outstanding commands per target port. The minimum value is 10. The maximum value is 65,535 (default). **NOTE:** The upper threshold 65,535 is determined by the adapter's capability.	Yes	All	Dynamic
lpfc_throttle_log_cnt	Specifies the number of messages logged within throttle_log_time. The minimum value is 1. The maximum value is 1000. The default value is 10.	Yes	All	Dynamic

Table 3: Dynamic FC Driver Parameters (Continued)

Parameter	Description	sysfs Visible	Applicable Adapters	Activation
lpfc_throttle_log_time	Specifies the time limit in seconds for throttle_log_cnt. The minimum value is 1. The maximum value is 60. The default value is 1.	Yes	All	Dynamic
lpfc_topology	For FC adapters, this parameter sets the link topology. The possible values are: ■ 0x0 = Loop first; if loop fails, then point-to-point (default). ■ 0x2 = Point-to-point only. ■ 0x4 = Loop only. ■ 0x6 = Point-to-point first; if point-to-point fails, then loop. **NOTE:** The lpfc_topology module parameter is not supported on LPe31000-series, LPe32000-series, and LPe35000-series adapters. On these adapters you can configure the topology using the Emulex HBA Manager application or the BIOS application. LPe31000-series and LPe32000-series adapters now support loop topology in addition to point-to-point.	Yes	LPe12000-series adapters	Port reset
lpfc_use_adisc	When enabled, an ADISC is sent instead of a PLOGI for device discovery or RSCN. The enabled value is 1. The default value is 0 (disabled).	Yes	All	Dynamic
lpfc_ras_fwlog_buffsize	For LPe31000-series, LPe32000-series, and LPe35000-series adapters, specifies the buffer size to be allocated in host memory for firmware logging. The possible values are: ■ 0 = Disables the firmware log written to the host memory (default). ■ 1 = ¼ MB (256K) host memory to be allocated for the firmware log. ■ 2 = ½ MB (512K) host memory to be allocated for the firmware log. ■ 3 = ¾ MB (768K) host memory to be allocated for the firmware log. ■ 4 = 1 MB host memory to be allocated for the firmware log.	Yes	LPe31000-series, LPe32000-series, and LPe35000-series adapters	Driver reload
lpfc_ras_fwlog_level	For LPe31000-series, LPe32000-series, and LPe35000-series adapters, configures the firmware logging level. Valid only if firmware logging is enabled. The minimum value is 0 (default). The maximum value is 4.	Yes	LPe31000-series, LPe32000-series, and LPe35000-series adapters	Driver reload

Table 3: Dynamic FC Driver Parameters (Continued)

Parameter	Description	sysfs Visible	Applicable Adapters	Activation
lpfc_nvmet_fb_size	For LPe31000-series, LPe32000-series, and LPe35000-series adapters, NVMe target mode supported first burst size. This is a target mode parameter, and when enabled the first burst size of the target is communicated to the NVMe initiator in the PRLI response. `lpfc_nvmet_fb_size` is used only when the `lpfc_nvme_enable_fb` is set and the target sends a PRLI response. The possible values are: 0 = Disabled (default) 1 = Enabled This parameter is supported only on physical ports.	Yes	LPe31000-series, LPe32000-series, and LPe35000-series adapters	Dynamic
lpfc_nvme_enable_fb	For LPe31000-series, LPe32000-series, and LPe35000-series adapters, enables NVMe first burst on initiator (I) and target (T) functions. For an initiator, enabling this parameter means that an NVMET PRLI response with FBA enabled and an FB_SIZE set to a nonzero value will be processed by the initiator for subsequent NVMe FCP I/O. Currently, this feature is not supported on NVMe targets. The possible values are: 0 = Disabled (default) 1 = Enabled	Yes	LPe31000-series, LPe32000-series, and LPe35000-series adapters	Dynamic
lpfc_nvme_oas	For LPe31000-series, LPe32000-series, and LPe35000-series adapters, use the oas bit when sending NVMe/NVMe target I/Os. Default value is 0 (disabled). The possible values are: 0 = Disabled (default) 1 = Enabled	Yes	LPe31000-series, LPe32000-series, and LPe35000-series adapters	Dynamic
lpfc_nvme_embed_cmd	This is an optional parameter. For LPe31000-series and LPe32000-series adapters, when enabled, embeds an NVMe command in WQE. The possible values are: 0 = Puts an NVMe command in SGL. 1 = Embeds an NVMe command in WQE (unless G7). 2 = Embeds an NVMe command in WQE (force).	Yes	LPe31000-series and LPe32000-series adapters	Dynamic

3.1.2 Configuring FC Driver Parameters

You can configure FC driver parameters using any of the following methods:

- The `modprobe` command for temporary configuration.
- The `elx-lpfc.conf` file for persistent configuration.
- The `sysfs` interface (to view and modify parameters after loading the FC driver).

NOTE: Not all parameters visible in the `sysfs` directory can be modified; some are read-only.

- The Emulex HBA Manager application (refer to the *Emulex HBA Manager User Guide* for more information).

NOTE: The FC driver parameter changes made using the Emulex HBA Manager application remain after the FC driver is uninstalled. To return to the default settings, you must reset them using the Emulex HBA Manager application.

3.1.2.1 Temporary Configuration with the modprobe Command

When you manually load the FC driver as a module using the `modprobe` command and you change one or more driver parameter values in the command line, the configuration is temporary. These changes are considered temporary because they are valid only for the current session or until the FC driver is unloaded.

The `modprobe` command uses the `elx-lpfc.conf` file, but parameters passed to it using the command line override the parameters in the `elx-lpfc.conf` file. Values can be expressed in hexadecimal or decimal notation.

If you want to temporarily set `lpfc_lun_queue_depth` to 20 (the default is 30) for all HBAs in your system, load the FC driver with the following command:

```
modprobe lpfc lpfc_lun_queue_depth=20
```

3.1.2.2 Persistent Configuration with the elx-lpfc.conf File

To make the FC driver parameters persist across module loads and reboots, perform the following steps:

1. In the `/etc/modprobe.d` directory, create a file with the driver name `elx-lpfc.conf`.
2. In `/etc/modprobe.d/elx-lpfc.conf`, use the `options` command to add the appropriate FC driver parameters and their desired values. For example, adding the following command to the `elx-lpfc.conf` file sets the verbose flag:

   ```
   options lpfc lpfc_log_verbose=0x3FFFF
   ```

If driver parameters are added to the `elx-lpfc.conf` file, the FC driver must be reloaded for the parameters to take effect. Also, a new ramdisk image is required if you want the changes to take effect in the next boot. For information on creating a new ramdisk, see Section 3.1.3, Creating a New Ramdisk Image.

If the same parameter is specified on the `modprobe` command line and in the `elx-lpfc.conf` file, the value specified in the `modprobe` command line takes precedence.

3.1.2.3 Configuring Parameters with a Read/Write to sysfs

`sysfs` is a virtual file system that exposes the structure of the system. It also includes interfaces to driver parameters through which the FC driver parameters can be viewed and modified. Because these interfaces are available only after driver load, only dynamic FC driver parameters can be changed. However, both static and dynamic FC driver parameters can be read through `sysfs`.

NOTE:

- `sysfs` changes exist only during driver load and are lost when the FC driver is unloaded or the system is rebooted.
- Driver parameters that are set through module parameters are global; setting them through `sysfs` is on a SCSI host (adapter port) basis.

  For example:

  `echo 0x7f > /sys/class/scsi_host/host7/lpfc_log_verbose` only affects host 7.

  `modprobe lpfc lpfc_log_verbose=0x7F` applies to all SCSI host (ports) managed by the lpfc driver.

Viewing Parameters with sysfs:

The `sysfs` file system is mounted and available as `/sys`. You must first identify the `scsi_host` that represents the adapter for which you want to modify the FC driver parameters. All `scsi_hosts` bound to the FC driver can be viewed with the following command:

```
ls -d /sys/bus/pci/drivers/lpfc/*/host*
```

Assuming that you are interested in adapter `scsi_host 7`, you can list the FC driver parameters for this particular adapter as follows:

```
ls -l /sys/class/scsi_host/host7/lpfc*
```

An example output follows:

```
-r--r--r--  1 root root 4096 Feb 28 17:03 /sys/class/scsi_host/host7/lpfc_ack0
-r--r--r--  1 root root 4096 Feb 28 17:03 /sys/class/scsi_host/host7/lpfc_fcp_class
-rw-r--r--  1 root root 4096 Feb 28 17:03 /sys/class/scsi_host/host7/lpfc_fdmi_on
-r--r--r--  1 root root 4096 Feb 28 17:03 /sys/class/scsi_host/host7/lpfc_link_speed
-rw-r--r--  1 root root 4096 Feb 28 15:34 /sys/class/scsi_host/host7/lpfc_log_verbose
-r--r--r--  1 root root 4096 Feb 28 17:03 /sys/class/scsi_host/host7/lpfc_lun_queue_depth
-rw-r--r--  1 root root 4096 Feb 28 17:03 /sys/class/scsi_host/host7/lpfc_max_luns
-rw-r--r--  1 root root 4096 Feb 28 17:03 /sys/class/scsi_host/host7/lpfc_devloss_tmo
-rw-r--r--  1 root root 4096 Feb 28 17:03 /sys/class/scsi_host/host7/lpfc_scan_down
-r--r--r--  1 root root 4096 Feb 28 17:03 /sys/class/scsi_host/host7/lpfc_topology
-rw-r--r--  1 root root 4096 Feb 28 17:03 /sys/class/scsi_host/host7/lpfc_use_adisc
```

Temporary Configuration Parameters with sysfs:

In the previous example, notice that the FC driver parameters are available as files. Reading a file displays the current value of a driver parameter. If the permissions allow it, you can write a value to the file and it will take effect immediately.

Reading the `lpfc_log_verbose` file might show that its value is `0`:

```
cat /sys/class/scsi_host/host7/lpfc_log_verbose
0
```

To modify the `lpfc_log_verbose` value to 0xffffffff:

```
echo 0xffffffff > /sys/class/scsi_host/host7/lpfc_log_verbose
```

Reading the lpfc_log_verbose file now shows a value of 0xffffffff:

```
cat /sys/class/scsi_host/host7/lpfc_log_verbose
0xffffffff
```

NOTE: Setting the lpfc_log_verbose value to 0xffffffff might degrade system performance.

3.1.3 Creating a New Ramdisk Image

The elx_lpfc_install.sh script creates a ramdisk image that contains the FC driver for the currently running kernel.

NOTE: You must create a new ramdisk image whenever the lpfc options in /etc/modprobe.conf are changed and you want the change to take effect on the next reboot.

To create a new initial ramdisk image for inbox FC drivers and installed binary RPM FC driver kits, type:

```
dracut -f /boot/initramfs-<kernel_version>.img <kernel_version>
```

3.1.4 Dynamically Recognizing LUNs and Targets (Using Scan)

The FC driver enables you to dynamically recognize LUNs and targets without unloading or reloading the FC module and without resetting the adapter.

To rescan an adapter's targets with sysfs, given the adapter's host number (in this example, 3), type:

```
echo "- - -" > /sys/class/scsi_host/host3/scan
```

To limit the rescan to a particular target, given the adapter's host number (in this example, 3) and the target number (in this example, 2), type:

```
echo "- 2 -" > /sys/class/scsi_host/host3/scan
```

You can also use the lun_scan script in the /usr/sbin/lpfc directory.

3.1.5 Persistent Naming

The generic device manager for the Linux kernel is *udev*, which primarily manages device nodes in the /dev directory.

3.1.5.1 Using udev to Discover Logical-to-Physical Mappings for sd Devices

In Linux, the driver for SCSI disk drives is sd. A disk device name has an sd prefix. Persistent names for sd devices are provided in the /dev/disk/by-id directory. To find the persistent udev name for the disk, which is currently sdc, type:

```
cd /dev/disk/by-id
ls -l | grep sdc
```

The sample output follows:

```
lrwxrwxrwx 1 root root 9 2006-08-01 19:08 scsi-32000000c5005d6e6 -> ../../sdc
```

In the previous example, the disk has no partitions. If the disk had two partitions, the output would look like the following:

```
lrwxrwxrwx 1 root root 9 2006-08-01 19:08 scsi-32000000c5005d6e6 -> ../../sdc
lrwxrwxrwx 1 root root 10 2006-08-01 19:08 scsi-32000000c5005d6e6-part1 -> ../../sdc1
lrwxrwxrwx 1 root root 10 2006-08-01 19:08 scsi-32000000c5005d6e6-part2 -> ../../sdc2
```

3.1.5.2 Configuring the System to Boot Using Persistent Names

For SLES Operating Systems:

NOTE: The SLES operating systems are configured by default with `udev` to provide persistent names for hard disks, including FC-attached disks.

To use a persistent name for a boot device:

1. Do one of the following:
   - For Legacy boot:

     In `/boot/grub/menu.lst`, find the kernel line for the default boot. For example:

     ```
     kernel /boot/vmlinuz root=/dev/sda2 vga=0x314
     ```
   - For UEFI boot:

     In /`boot/efi/EFI/sles/grub.cfg`, find the kernel line for the default boot. For example:

     ```
     kernel /boot/vmlinuz root=/dev/sda2 vga=0x314
     ```
2. Find the persistent name for the root partition (following `root=` on the kernel line) by using the instructions in Section 3.1.5.1, Using udev to Discover Logical-to-Physical Mappings for sd Devices.
3. In the same file on Legacy boot or UEFI boot, replace the text after `root=` with the partition's persistent name. For example:

   ```
   kernel /boot/vmlinuz root=/dev/disk/by-id/scsi-32000000c5005d6e6-part2 vga=0x314
   ```
4. Change any mounts listed in `/etc/fstab` that refer to this root partition by either its `/dev/sd` name or a file system label to use the persistent name as well.

For the RHEL Operating Systems:

To use a persistent name for a boot device:

1. Do one of the following:
   - For Legacy boot:

     In `/boot/grub/grub.conf`, find the kernel line for the default boot. For example:

     ```
     kernel /boot/vmlinuz -<kernel version> ro root=/dev/sda2
     ```
   - For UEFI boot:

     In `/boot/efi/EFI/redhat/grub.cfg`, find the kernel line for the default boot. For example:

     ```
     kernel /boot/vmlinuz -<kernel version> ro root=/dev/sda2
     ```
2. Find the persistent name for the root partition (following `root=` on the kernel line) by using the instructions in Section 3.1.5.1, Using udev to Discover Logical-to-Physical Mappings for sd Devices.
3. In the same file depending on Legacy boot or UEFI boot, replace the text after `root=` with the partition's persistent name. For example:

   ```
   kernel /boot/vmlinuz -<kernel version> ro root=/dev/disk/by-id/scsi-32000000c5005d6e6-part2
   ```
4. Change any mounts listed in `/etc/fstab` that refer to this root partition by either its `/dev/sd` name or a file system label to use the persistent name as well.

3.1.5.3 Using udev with st Devices

In Linux, the driver for SCSI tape drives is `st`. A tape device name has an `st` prefix. The `udev` rules for tape devices are the same as for disk devices. A unique ID must exist that persists across initiator reboots and persists regardless of discovery order.

You must consider whether the tape device is an FC tape device or an FC-SCSI tape device (in which multiple SCSI tape devices reside behind an FC controller). If it is an FC tape device, the WWPN is unique and can be used to create the persistent name. In this case, the `scsi_id` command should return the WWPN as the unique identifier with a single-digit prefix. If the FC controller has multiple SCSI tape devices behind it, the WWPN is not unique, and the persistent name must use multiple information elements to build the unique ID. FC Tape Device Examples and FC-SCSI Tape Device Example are examples of each scenario.

FC Tape Device Examples:

The following examples use the `scsi_id` command to retrieve and generate a unique SCSI identifier:

```
scsi_id [options]
```

For these examples, the following [*options*] are used:

- `-g` Treats the device as white-listed. It is needed on the command line or in the `scsi_id.config` file for the `scsi_id` command to generate any output. In the examples, the `-g` option is needed on the command line because the vendor and model for this tape device are not in the `/etc/scsi_id.config` file.
- `-s` Generates an ID for the `sysfs` device. Note that `-s` is an invalid option for `scsi_id` version 147.

NOTE: Because the [*options*] can vary depending on the version of the `scsi_id` command, refer to the `scsi_id` man page on your system for the correct and complete list of [*options*].

The following example is an FC tape device using the SCSI generic driver (`sg`) rather than the SCSI tape driver. The value returned has a leading prefix of 3, which is the NAA type. The remaining digits represent the FC controller's WWPN.

```
scsi_id -g -s /sys/class/scsi_generic/sg0
350060b000029b592
```

The following example is an FC tape device using the SCSI tape driver. The value returned is the same as the previous example.

```
scsi_id -g -s /sys/class/scsi_tape/nst0
350060b000029b592
```

The following example uses a different FC tape vendor. Notice that the value returned is similar to the previous examples, with respect to the leading digit and the WWPN.

```
/sbin/scsi_id -g -s sys/class/scsi_tape/nst0
35005076300015101
```

FC-SCSI Tape Device Example:

The following is an example of an FC controller with multiple SCSI tape devices behind it (FC-SCSI tape device). When the Emulex driver is loaded, the SCSI mid-level discovers the SCSI tape devices as follows:

```
scsi scan: INQUIRY to host 14 channel 0 id 0 lun 0
scsi: unknown device type 12
Vendor: ADIC      Model: SNC 4000          Rev: 42d4
Type:   RAID                               ANSI SCSI revision: 03
Attached scsi generic sg5 at scsi14, channel 0, id 0, lun 0,  type 12
scsi scan: INQUIRY to host 14 channel 0 id 0 lun 1
Vendor: ADIC      Model: Scalar 24         Rev: 227A
Type:   Medium Changer                     ANSI SCSI revision: 02
```

```
Attached scsi generic sg6 at scsi14, channel 0, id 0, lun 1,type 8
scsi scan: INQUIRY to host 14 channel 0 id 0 lun 2
Vendor: IBM       Model: ULTRIUM-TD2       Rev: 38D0
Type:   Sequential-Access                  ANSI SCSI revision: 03
Attached scsi tape st0 at scsi14, channel 0, id 0, lun 2
st0: try direct i/o: yes (alignment 512 B), max page reachable by HBA 4503599627370495
Attached scsi generic sg7 at scsi14, channel 0, id 0, lun 2,  type 1
scsi scan: INQUIRY to host 14 channel 0 id 0 lun 3
Vendor: IBM       Model: ULTRIUM-TD2       Rev: 38D0
Type:   Sequential-Access                  ANSI SCSI revision: 03
Attached scsi tape st1 at scsi14, channel 0, id 0, lun 3
st1: try direct i/o: yes (alignment 512 B), max page reachable by HBA 4503599627370495
Attached scsi generic sg8 at scsi14, channel 0, id 0, lun 3,  type 1
```

This log output shows a controller at LUN 0, the medium changer at LUN 1, and two SCSI tape devices at LUNs 2 and 3.

The following example is the result of a `scsi_id` call:

```
scsi_id -g -s /sys/class/scsi_tape/nst0
1IBM     ULTRIUM-TD2     1110133831
scsi_id -g -s /sys/class/scsi_tape/nst1
1IBM     ULTRIUM-TD2     1110133994
```

Notice that the unique ID contains three values with space delimiters. A `udev` rule must have a unique ID for the device, so that all three parts of this returned string are required. To do this, use the following command:

```
scsi_id -u -g -s /sys/class/scsi_tape/nst0
1IBM_____ULTRIUM-TD2_____1110133831
scsi_id -u -g -s /sys/class/scsi_tape/nst1
1IBM_____ULTRIUM-TD2_____1110133994
```

Creating the udev Persistent Name for SCSI Tape Device:

After you know the SCSI ID call needed to extract a unique ID, use the same process to create a `udev` persistent name for a SCSI tape device as on a SCSI disk device.

The rule for the FC tape device follows:

```
BUS="scsi", SYSFS{vendor}="HP", SYSFS{model}="ULTRIUM 3-SCSI", PROGRAM="/sbin/scsi_id -p 0x83 -u
-g -s /sys/class/scsi_tape/nst%n",RESULT="350060b000029b592", SYMLINK="fc_lun_st%n"
```

The rule for the FC-SCSI tape device follows:

```
BUS="scsi", SYSFS{vendor}="IBM", SYSFS{model}="ULTRIUM-TD2", PROGRAM="/sbin/scsi_id -p 0x83 -u -g
-s /sys/class/scsi_tape/nst%n",RESULT="1IBM_____ULTRIUM-TD2_____1110133831",
SYMLINK="fc_lun_st%n"
BUS="scsi", RESULT="1IBM_____ULTRIUM-TD2_____1110133994", SYMLINK="fc_lun_st%n"
```

Create a new file named `/etc/udev/rules.d/45-local.rules`,and put the appropriate rule in it. Then run `udevtrigger` to reload the `udev` rules, and the output of the rules will follow:

```
udevtrigger
ls -al /dev/fc*
lrwxrwxrwx  1 root root 3 Apr  7 15:03 fc_lun_st0 -> st0
lrwxrwxrwx  1 root root 3 Apr  7 15:03 fc_lun_st1 -> st1
```

3.1.5.4 Persistent Naming References

See the following references for more information on persistent naming:

- www.reactivated.net/udevrules.php by Daniel Drake (dsd)
- kernel.org/pub/linux/utils/kernel/hotplug/udev_vs_devfs by Greg Kroah-Hartman

3.1.6 ExpressLane Support

The host servers do not distinguish between lower and higher priority workloads being sent to LUNs. For flash storage deployed in mixed storage environments, the combination of data from rotating media and flash devices can cause congestion on the network and reduced performance for the flash storage devices.

The ExpressLane feature enables special priority queuing for selected LUNs (ExpressLane LUNs). The ExpressLane LUN performance is superior to that of regular LUNs. Mixed storage environments can use the ExpressLane feature to alleviate congestion, reduce latency, and improve throughput, ensuring that key applications receive highest priority. Switches can provide up to three priority levels: Low, Medium, and High,

The following requirements must be met to use the ExpressLane feature:

NOTE: The ExpressLane feature is supported only on Emulex LPe16000-series, LPe31000-series, and LPe32000-series adapters.

An ExpressLane LUN must be enabled in the driver before it can be used by the Emulex HBA Manager application. Additionally, the priority value to mark each of the ExpressLane frames must be specified to the FC/FCoE driver through the appropriate driver parameters. For additional information, refer to the *Emulex HBA Manager User Guide*.

NOTE: Enabling the ExpressLane feature on a LUN, when the maximum number of LUNs are already enabled for ExpressLane, might result in an error. Use the `/usr/sbin/lpfc/lpfc_clean_xlane_conf.sh` script to clear any unwanted entries and retry enabling ExpressLane.

3.2 vPort Configuration

This section describes how to create, delete, and display vPorts. You can also configure the vPorts using the Emulex HBA Manager application. For more information, refer to the *Emulex HBA Manager User Guide*.

3.2.1 vPort Configuration Prerequisites

Before configuring vPorts, consider the following points:

- Ensure that you are using the latest firmware for vPort functionality. For the latest firmware, go to www.broadcom.com.

NOTE: Before performing a firmware update, driver installation is required. For more information on installing the driver, see Chapter 2, Installing and Uninstalling.

- Loop devices and NPIV are not supported on the same port simultaneously. If you are running a loop topology and you create a vPort, the vPort link state is offline.
- vPorts do not persist across system reboots.

3.2.2 Creating, Deleting, and Displaying vPorts

vPorts are created through sysfs entries that are presented in the physical port's sysfs directory. There are three scripts to create, delete, and display vPorts. The scripts reside in the /usr/sbin/lpfc directory and are part of the Emulex HBA Manager application kit.

When NPIV is enabled and vPorts are configured, it can take longer for the adapter to finish discovery in some cases because each vPort must perform discovery independently. As more vPorts are configured, the amount of time that the driver and adapter take to finish discovery of remote ports on the SAN increases. To compensate for this extended amount of time taken during discovery, set the lpfc_devloss_tmo parameter to 60 seconds when NPIV is enabled.

3.2.2.1 Creating vPorts Using the mkvport.sh Script

You can use the mkvport.sh script to create vPorts. To see the usage information, run the script with no parameters specified. The mkvport.sh script uses the following syntax:

```
./mkvport.sh <physical_port's_host_number> <port_name> <node_name>
```

You must supply the physical port’s host number, WWPN, and WWNN when using the mkvport.sh script. For example, to create a vPort with port name of 10000000c94ac63a and a node name of 20010000c94ac63a on the physical port with scsi_host name host7, type:

```
./mkvport.sh host7 10000000c94ac63a 20010000c94ac63a
```

This script fails if the vPort is not created.

NOTE: It is possible for a vPort to be created successfully but to be in a failed state. For example, loop devices and NPIV are not supported on the same port simultaneously. If you are running a loop topology and you create a vPort, the vPort’s link state will be offline.

3.2.2.2 Deleting vPorts Using the rmvport.sh Script

NOTE: You must unmap, unmount, and flush I/Os to vPort-connected devices before deleting the vPort.

You can use the rmvport.sh script to delete vPorts. To see the usage information, run the script with no parameters specified. The rmvport.sh script uses the following syntax:

```
./rmvport.sh <vPort_host_number>
```

Or

```
./rmvport.sh <port_name> <node_name>
```

To delete the vPort with a port name of 10000000c94ac63a and a node name of 20010000c94ac63a, type:

```
./rmvport.sh 10000000c94ac63a 20010000c94ac63a
```

This script can take up to 30 seconds to finish. The script fails if the vPort is not deleted.

3.2.2.3 Displaying vPorts Using the lsvport.sh Script

You can use the `lsvport.sh` script to display the vPorts and physical ports that are present on the system. Run the script with no parameters to display port information.

For example:

```
./lsvport.sh
lpfc0: host6 10000000c93a5b5e:20000000c93a5b5e LP10000 NPIV Not Supported
lpfc1: host7 10000000c93a5b5d:20000000c93a5b5d LP10000 NPIV Not Supported
lpfc2: host8 10000000c93cc8dd:20000000c93cc8dd LPe12000 NPIV Physical
lpfc3: host9 10000000c93cc8dc:20000000c93cc8dc LPe12000 NPIV Physical
lpfc4: host10 10000000c94ac63a:20010000c94ac63a  NPIV Virtual (VPI 1)
```

In reference to the previous example:

- For LPFC0 and LPFC1, `NPIV Not Supported` indicates that this adapter/firmware combination does not support the creation of vPorts.
- For LPFC2 and LPFC3, `NPIV Physical` refers to a physical port of this adapter.
- For LPFC4, `NPIV Virtual` refers to a vPort of this adapter.

3.2.3 vPort sysfs

This section describes the vPort `sysfs` structure and vPort `sysfs` entries.

3.2.3.1 vPort sysfs Tree

When a vPort is created, three new directories are created in the class tree:

```
/sys/class/scsi_host/hostY/
/sys/class/fc_host/hostY/
/sys/class/fc_vports/vport-X:0-Z/
```

Creating a new vPort also creates a new `sysfs` directory in the bus and devices tree:

```
/sys/bus/pci/drivers/lpfc/0000:A:B:C/hostX/vport-X:0-Z/hostY
/sys/devices/pci0000:A/0000:A:B:C/hostX/vport-X:0-Z/hostY
```

In both directories, a `hostY` directory contains the remote ports that this new host can access:

```
/sys/bus/pci/drivers/lpfc/0000:A:B:C/hostX/vport-X:0-Z/hostY
/sys/bus/pci/drivers/lpfc/0000:A:B:C/hostX/vport-X:0-Z/hostY/rport-Y:0-0
/sys/bus/pci/drivers/lpfc/0000:A:B:C/hostX/vport-X:0-Z/hostY/rport-Y:0-1
/sys/bus/pci/drivers/lpfc/0000:A:B:C/hostX/vport-X:0-Z/hostY/rport-Y:0-2
```

In this example:

- `X` indicates the host value for the parent `fc_host` from which this vPort was created.
- `Y` indicates the new host value for the vPort that was created.
- `Z` indicates the instance of vPort created from the parent `fc_host`. `A`, `B`, and `C` indicate the PCI hierarchy for each physical FC port.

`hostY` is the new host created for the new vPort. `vport-X:0-Z` uniquely identifies the vPort and indicates the parent host structure (`X`) that created this vPort.

For example, if a vPort is created from host5, a new scsi_host, a new fc_host, a new fc_vport, and a new entry under the bus tree are created as well.

```
ls /sys/class/scsi_host/
host0  host1  host4  host5  host6
ls /sys/class/fc_host/
host4  host5  host6
ls /sys/class/fc_vports/
vport-5:0-0
```

3.2.3.2 vPort sysfs Entries

NOTE: vPort sysfs entries shown in the following table are present only if the driver was loaded with lpfc_enable_npiv enabled.

Table 4: vPort sysfs Entries

vPort sysfs Entries	Type	Range/Input	Location and Description
lpfc_peer_port_login	Read/Write	0 = Off (default) 1 = On	/sys/class/scsi_host/hostX/ lpfc_peer_port_login This entry sets the port behavior when discovering targets in the SAN. The default behavior (value = 0) logs in only to N_Ports that are physically located on a different port. The port still attempts to log in to targets on all other ports (including the other port in a dual port adapter). If this parameter is turned on (value = 1), the port attempts to log in to all N_Ports, even if they are physically located on the same port. **NOTE:** This parameter was created to reduce the number of hardware resources (for example, RPIs) that the driver requires. In a configuration where there are many vPorts on one physical port, this capability greatly reduces the number of RPIs that the driver uses.
lpfc_restrict_login	Read/Write	0 = Off 1 = On (default)	/sys/class/scsi_host/hostX/ lpfc_restrict_login (vPorts only) This entry sets the vPort behavior when discovering targets in the SAN. The default behavior (value = 1) prevents the vPort from logging in to other initiator ports in the SAN. It also rejects logins from other ports in the SAN because it assumes that all ports that send a PLOGI are initiators. If this sysfs entry is turned off, the driver attempts to log in to every port that it can access in the SAN, and it accepts logins from all ports. **NOTE:** This parameter was created to reduce the number of hardware resources (for example, RPIs) that the driver requires. In a SAN where other initiators exist, this capability greatly reduces the number of RPIs that the driver uses.

Table 4: vPort sysfs Entries (Continued)

vPort sysfs Entries	Type	Range/Input	Location and Description
max_npiv_vports	Read-only	integers	/sys/class/fc_host/hostX/max_npiv_vports This entry displays the maximum number of vPorts that are supported by the underlying fc_host hardware. This sysfs entry exists only if the vport_create and vport_delete sysfs entries exist. If an fc_host does not support NPIV, this sysfs entry might not exist. Use this sysfs entry with the npiv_vports_inuse entry to determine whether the maximum number of vPorts have been created on this fc_host.
node_name	Read-only	16-byte hexadecimal value	/sys/class/fc_host/hostX/node_name /sys/class/fc_vports/vport-X:0-Z/node_name These entries display the physical or vPort node name. You assign this value when the vPort is created, and it is transmitted to the fabric upon fabric login.
npiv_vports_inuse	Read-only	integers	/sys/class/fc_host/hostX/npiv-_vports_inuse This entry displays the number of vPorts that were created on this fc_host. This sysfs entry exists only if the vport_create and vport_delete sysfs entries exist. If an fc_host does not support NPIV, this sysfs entry might not exist. Use this sysfs entry with max_npiv_vports to determine whether the maximum number of vPorts have been created on this fc_host.
port_name	Read-only	16-byte hexadecimal value	/sys/class/fc_host/hostX/port_name /sys/class/fc_vports/vport-X:0-Z/port_name This entry displays the physical or vPort port name. You assign this value when the vPort is created, and it is transmitted to the fabric upon fabric login.

Table 4: vPort sysfs Entries (Continued)

vPort sysfs Entries	Type	Range/Input	Location and Description
vport_create	Write-only	WWPN; WWNN	/sys/class/fc_host/hostX/vport_create This entry creates a vPort on the physical port on which hostX is located. The new vPort has a WWPN and WWNN present on the fabric based on the WWPN and WWNN that are entered with this sysfs entry. This entry returns a 0 if the vPort creation was successful. A nonzero value indicates that the vPort was not created. If an fc_host does not support NPIV, this sysfs entry might not exist. **NOTE:** It is possible for the vPort creation to succeed, fail, or be in an inoperative state. Use the new sysfs tree created by the new vPort to check the state of the new vPort.
vport_delete	Write-only	WWPN; WWNN	/sys/class/fc_host/hostX/vport_delete This entry deletes a vPort on the physical port on which hostX is located. The vPort that matches the WWPN and WWNN is immediately deleted. This entry returns a 0 if the vPort deletion was successful. A nonzero value indicates that the vPort was not deleted. If an fc_host does not support NPIV, this sysfs entry might not exist. **NOTE:** This entry deletes the vPort even if there are mounted file systems being accessed through this vPort or if there are open files on it.

3.2.4 Monitoring vPorts with fc_vport

The transport creates an fc_vports directory that you can use to monitor vPorts. This directory is populated entirely of vPorts and has links from each to the fc_host associated with that vPort.

```
ls /sys/class/fc_vports/
vport-5:0-0
ls -d /sys/bus/pci/drivers/lpfc/*/host*/*/host*
/sys/bus/pci/drivers/lpfc/0000:03:06.1/host5/vport-5:0-0/host6
ls /sys/devices/pci*/*/host5/vport-5*/host6
power rport-6:0-0 rport-6:0-1 rport-6:0-2 uevent
ls /sys/devices/pci*/*/host5/vport-5*/host6/rport-*
/sys/devices/pci00:03/00:03:06.1/host5/vport-5:0-0/host6/rport-6:0-0:
power uevent
/sys/devices/pci00:03/00:03:06.1/host5/vport-5:0-0/host6/rport-6:0-1:
power uevent
/sys/devices/pci00:03/00:03:06.1/host5/vport-5:0-0/host6/rport-6:0-2:
power target6:0:0 uevent
```

In this example:

- A new entry exists in the `fc_vports` directory for the vPort (`vport-5:0-0`). The `vport-5:0-0` entry indicates that the vPort was created from `host5`, and it is the first (0) vPort to be created on that `fc_host`.
- The new host for the vPort is `host6`, and it appears in the usual directories.
- A new directory exists in the bus tree. This new directory indicates that `host6` was created under `vport-5:0-0` (which was created from `host5`).

3.2.5 vPort Configuration Limits

vPort configuration limits are designated as enforced or unenforced. Enforced limits are limits that the driver enforces and that you cannot exceed. Unenforced limits are limits that the driver cannot enforce, but configurations that exceed them are unsupported.

The following vPort configuration limits have been tested with and are supported by the Emulex driver. Configurations that exceed one or more of these limits are unsupported.

NOTE: Before the vPort is deleted or the driver is unloaded, I/O devices accessed through a vPort must be stopped and file systems must be unmounted.

- The maximum number of vPorts configurable on a physical port is 255.
- The maximum number of LUNs supported on each driver port is 256.
- The maximum number of targets supported for each driver port is 255.
- The maximum number of driver ports in one zone is 64. This limit is based on the system’s ability to recover from link events within the time constraints of the default timers.

The NPIV use cases that involve a virtual server environment include associating a vPort with a virtual machine, and placing the virtual machine in its own zone, which results in one vPort per zone. In the case of load-balanced environments, this can increase typically to two vPorts per virtual machine, to a practical limit of something far less than 50.

In the NPIV cases not related to virtual server environments, zoning is typically initiator-zoning, again resulting in one vPort, or a low number of vPorts in the case of load balancing, within a given zone. If too many vPorts exist within a single zone, expected behavior includes devices being lost after link events.

The minimum lifetime of a vPort is 60 seconds. An unenforced limit of 60 seconds exists between the creation of a vPort and the deletion of the same vPort. vPorts are designed to exist for a long time in the system, and the creation of vPorts is asynchronous, which means that a vPort might not be finished with FC or SCSI discovery when the command to create a vPort is finished.

3.3 FC Driver Performance Tuning

This section describes how to tune the FC driver for best performance.

3.3.1 Overview

The configurable parameters `lpfc_fcp_io_sched` and `lpfc_fcp_imax` can enhance performance on RHEL 7.5 and earlier kernels.

The configurable parameters `lpfc_hdw_queue` and `lpfc_irq_chann` can enhance performance on the following operating systems:

- RHEL 8.0 and later
- RHEL 7.6 and later
- SLES 15 and later
- SLES 12 and later

These features are available through module parameters that are defined in the FC driver as well as `sysfs` entries defined by the Linux kernel.

This section provides more information about how the tuning parameters and script can improve Emulex adapter performance.

NOTE: The parameters in this section do not apply to LPe12000-series adapters.

3.3.1.1 lpfc_fcp_io_sched

The `lpfc_fcp_io_sched` module parameter can be configured at driver load time. It can also be set dynamically as an `sysfs` entry. The driver uses the parameter to determine which algorithm to use when scheduling an FCP I/O to an I/O channel.

When multiple I/O channels are in use, I/Os can be scheduled to an I/O channel in a round-robin fashion or by determining which CPU is running when the I/O is submitted.

The default value (0) configures the driver for round-robin scheduling. A value of 1 configures scheduling by CPU.

NOTE: This parameter is not applicable for LPe12000-series adapters, and by default, the value is configured to CPU scheduling.

3.3.1.2 lpfc_fcp_imax

The `lpfc_fcp_imax` parameter can be configured at driver load time. It also can be set dynamically as an `sysfs` entry. This parameter defines the maximum number of interrupts per second that each adapter port will support.

Considerations:

- The lower the value set, the more completions are coalesced by the adapter, which causes the driver to handle multiple completions under the context of one interrupt. The higher the value, the faster an interrupt is generated for a completed command. Therefore, a balanced or *tuned* system must be found.
- A lower value equals higher interrupt latency; a higher value equals lower interrupt latency.
- Faster completions consume more system resources and CPU cycles, as the overhead of one interrupt completes fewer commands.

3.3.1.3 lpfc_hdw_queue

The `lpfc_hdw_queue` module parameter can be configured at driver load time. It defines the number of hardware queues supported by the driver for each port. The driver is capable of supporting parallel I/O paths, and each I/O path is capable of posting and completing FCP/NVMe commands independent of the other. Each hardware queue is composed of a unique CQ/WQ pair.

NOTE: The Emulex LPe12000-series adapters support only one I/O path, so this parameter has no effect on them.

By default, `lpfc_hdw_queue` is configured for the number of online logical CPUs per port. The driver also limits the number of hardware queues to not exceed the number of online logical CPUs (as reported by `/proc/cpuinfo`). It is highly desirable, for performance, to have one hardware queue per CPU.

3.3.1.4 lpfc_irq_chann

The `lpfc_irq_chann` module parameter can be configured at driver load time. It defines the number of MSI-X vectors supported by the driver for each port. The driver is capable of supporting parallel I/O paths, and each I/O path is capable of posting and completing FCP/NVMe commands independent of the other.

Each IRQ channel is composed of a unique MSI-X vector and an EQ.

NOTE: The Emulex LPe12000-series adapters support only one I/O path, so this parameter has no effect on them.

By default, `lpfc_irq_chann` is configured for the number of online logical CPUs per port. The driver will also limit `lpfc_irq_chann` to not exceed the number of online logical CPUs (as reported by `/proc/cpuinfo`) and to not exceed `lpfc_hdw_queue`. If `lpfc_irq_chann` is less then `lpfc_hdw_queue` multiple hardware queues must share the same IRQ channel. It is highly desirable, for performance, to have one IRQ channel per hardware queue.

Chapter 4: Configuring NVMe over FC

NVMe over FC is supported on the following operating systems:

- RHEL 8.0 and later
- RHEL 7.6 and later
- SLES 15 and later
- SLES 12 SP4 and later

For supported operating systems, see Section 2.6, Installing the NVMe over FC Driver.

To configure NVMe over FC, you must perform the following procedures in the indicated sequence:

1. Attach NVMe over FC-based storage to the FC SAN. Alternatively, an Emulex adapter may be configured as an NVMe over FC target and used by the Linux NVMeT subsystem (see Appendix B, Configuring NVMe over FC on a Target).
2. Configure NVMe over FC on initiator systems, as described in this chapter.

NOTE:

- The `nvmefc-connect` RPM is not installed by default. To install the RPM, the `elx_lpfc_install.sh` script includes an `-n` option. When the `-n` option is used, the install script installs the driver RPM and the `nvmefc-connect` RPM. The `nvmefc-connect` RPM is not needed on SUSE systems. It is required on all other operating systems.
- All operating system distributions will be releasing the upstream version of the NVMe over fabrics auto-connect scripts, which will replace this lpfc RPM.
- Before configuring NVMe over FC using native NVMe CLI commands, ensure that you have installed the latest Emulex firmware for the LPe31000-series, LPe32000-series, and LPe35000-series adapters.
- The target system must be configured and running before you connect the host system to the target.

4.1 Configuring an NVMe over FC Initiator Using Native CLI Commands

NVMe over FC on an initiator is configured automatically with the lpfc installer script `elx_lpfc_install.sh`.

To install the lpfc driver with NVMe on the initiator, perform the following steps:

1. Type the following command:

   ```
   # elx_lpfc_install.sh -n
   ```

   `-n` installs the `nvmefc-connect` RPM if the running operating system is supported.

NOTE:

- The `nvmefc-connect` RPM is not needed on SUSE systems. It is required on all other operating systems.
- The existing lpfc driver parameters in the `elx-lpfc.conf` file are not affected.
- By default, the SCSI and NVMe protocols are enabled on the port; and by default, the port is an initiator for the SCSI/NVMe protocol.

2. Reboot the system.

   After the system is rebooted, the configured protocols are enacted on the FC ports.

   The `nvmefc-connect.rpm` file, which is installed with the out-of-box Linux driver, automatically causes the initiator to connect to the target devices. Connections are established whenever any of the following events occur, as long as the target is running:

   – A host system reboot
   – A linkup event on the initiator port
   – A fabric zone event

   After configuring the initiator, you can run the `nvme_info` command to verify that your configuration is working by typing the following command on the initiator:

   ```
   cat /sys/class/scsi_host/host<X>/nvme_info
   ```

   where <*X*> is the host number.

   Information similar to the following is displayed:

   ```
   NVME Initiator Enabled
   XRI Dist lpfc0 Total 6144 IO 5894 ELS 250
   NVME LPORT lpfc0 WWPN x10000090fa9429e4 WWNN x20000090fa9429e4 DID x011d00 ONLINE
   NVME RPORT       WWPN x10000090fa942f0c WWNN x20000090fa942f0c DID x010a00 TARGET DISCSRVC ONLINE

   NVME Statistics
   LS: Xmt 0000000348 Cmpl 0000000348 Abort 00000000
   LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
   Total FCP Cmpl 00000000000019cb Issue 00000000000019cf OutIO 0000000000000004
          abort 00000000 noxri 00000000 nondlp 00000000 qdepth 00000000 wqerr 00000000 err 00000000
   FCP CMPL: xb 00000010 Err 00000015
   ```

3. If multipathing is used in SLES 15 or SLES 15 SP1, skip to Step 4. For operating systems with a single path, use the `nvme list` command to display target connections by typing the following command on the initiator:

   ```
   nvme list
   ```

NOTE: There are two similar commands to list all NVMe namespace devices:

- `nvme list` (with a space) is a native NVMe CLI command.
- `nvme-list-linux` (with a hyphen) is an Emulex HBA Manager CLI command.

This chapter refers to the `nvme list` command.

Information similar to the following is displayed:

```
Node             SN                   Model  Namespace  Usage                      Format           FW Rev
---------------- -------------------- ------ ---------- -------------------------- ---------------- ---------
/dev/nvme0n1     981fadb118adb0fa     Linux  1          268.44  GB / 268.44  GB    512   B +  0 B   4.4.70-2
/dev/nvme0n2     981fadb118adb0fa     Linux  2          268.44  GB / 268.44  GB    512   B +  0 B   4.4.70-2
```

Skip to Step 5.

4. For multipathing in SLES 15 or SLES 15 SP1, type the following command to display target connections on the initiator:

   ```
   nvme list-subsys
   ```

Information similar to the following is displayed:

```
nvme-subsys0 - NQN=nqn.2014-08.org.nvmexpress:NVMf:uuid:19d48a6d-3644-4f65-8046-80bd88fe45a6
\
 +- nvme0 fc traddr=nn-0x20000090fa942e16:pn-0x10000090fa942e16 host_traddr=nn-0x200000109b346f1f:pn-
0x100000109b346f1f
 +- nvme3 fc traddr=nn-0x20000090fa942e16:pn-0x10000090fa942e16 host_traddr=nn-0x200000109b346f20:pn-
0x100000109b346f20
nvme-subsys1 - NQN=nqn.2014-08.org.nvmexpress:NVMf:uuid:2c6d862f-b77e-43b1-b345-7f990125088e
\
 +- nvme1 fc traddr=nn-0x20000090fa942e16:pn-0x10000090fa942e16 host_traddr=nn-0x200000109b346f1f:pn-
0x100000109b346f1f
 +- nvme6 fc traddr=nn-0x20000090fa942e16:pn-0x10000090fa942e16 host_traddr=nn-0x200000109b346f20:pn-
0x100000109b346f20
nvme-subsys2 - NQN=nqn.2014-08.org.nvmexpress:NVMf:uuid:4bbbd877-6de2-4a67-ad56-e0333e2437b5
\
 +- nvme2 fc traddr=nn-0x20000090fa942e16:pn-0x10000090fa942e16 host_traddr=nn-0x200000109b346f1f:pn-
0x100000109b346f1f
 +- nvme7 fc traddr=nn-0x20000090fa942e16:pn-0x10000090fa942e16 host_traddr=nn-0x200000109b346f20:pn-
0x100000109b346f20
nvme-subsys3 - NQN=nqn.2014-08.org.nvmexpress:NVMf:uuid:c8da7263-fefe-4563-a14c-a9f0362dcb03
\
 +- nvme5 fc traddr=nn-0x20000090fa942e16:pn-0x10000090fa942e16 host_traddr=nn-0x200000109b346f1f:pn-
0x100000109b346f1f
 +- nvme9 fc traddr=nn-0x20000090fa942e16:pn-0x10000090fa942e16 host_traddr=nn-0x200000109b346f20:pn-
0x100000109b346f20
nvme-subsys4 - NQN=nqn.2014-08.org.nvmexpress:NVMf:uuid:5f1d7788-fab1-45cf-af9c-d30dc9a9ce89
\
 +- nvme4 fc traddr=nn-0x20000090fa942e16:pn-0x10000090fa942e16 host_traddr=nn-0x200000109b346f1f:pn-
0x100000109b346f1f
 +- nvme8 fc traddr=nn-0x20000090fa942e16:pn-0x10000090fa942e16 host_traddr=nn-0x200000109b346f20:pn-
0x100000109b346f20
```

NOTE: For additional information on multipathing, see Appendix C, Native NVMe Multipathing.

5. You can run the `lsblk` command on the initiator to display all the block devices, including newly discovered ones, by typing the following command:

   ```
   lsblk
   ```

   Information similar to the following is displayed:

```
  NAME                                          MAJ:MIN  RM   SIZE     RO   TYPE     MOUNTPOINT
sda                                               8:0    0    136.7G   0    disk
├─sda1                                            8:1    0    2G       0    part
├─sda2                                            8:2    0    40G      0    part
├─sda3                                            8:3    0    94.7G    0    part
└─3600508b1001037383941424344450400             254:0    0    136.7G   0    mpath
  ├─3600508b1001037383941424344450400-part1     254:1    0    2G       0    part     [SWAP]
  ├─3600508b1001037383941424344450400-part2     254:2    0    40G      0    part     /var/opt
  └─3600508b1001037383941424344450400-part3     254:3    0    94.7G    0    part     /home
sdb                                              8:16    0    3G       0    disk
  └─3600000e00d2a0000002a017300340000           254:4    0    3G       0    mpath
sdc                                              8:32    0    3G       0    disk
  └─3600000e00d2a0000002a017300350000           254:5    0    3G       0    mpath
```

```
sdd                                                    8:48   0    3G        0    disk
  └─3600000e00d2a0000002a017300360000                  254:6   0    3G        0    mpath
nvme0n1                                               259:0   0    3G        0    disk
nvme0n2                                               259:1   0    3G        0    disk
```

NOTE: The `/dev/sde` and `/dev/sdf` block devices, which were previously referred to on the NVMe target system, are now discovered by the NVMe initiator system, and they appear as the NVMe block devices `/dev/nvme0n1` and `/dev/nvme0n2`.

4.2 Emulex Support for NVMe CLI Commands

The following operating systems include many native NVMe CLI commands for initiators:

- RHEL 8.0 and later
- RHEL 7.6 and later
- SLES 15 and later
- SLES 12 SP4 and later

Of these, Emulex has tested, and thus supports, only the following commands at this time:

- `list`
- `connect-all`
- `connect`
- `gen-hostnqn`
- `list-subsys` (SLES 15 and SLES 15 SP1 only)

Documentation for these commands is available with the operating system.

Chapter 5: Troubleshooting

This chapter explains some situations in which your system can operate in an unexpected manner and some possible resolutions.

5.1 Hardware Situations and Resolutions

Table 5 lists the hardware situations and their resolutions.

Table 5: Hardware Situations and Their Resolutions

Situation	Resolution
An unapproved optical transceiver is used to connect the adapter.	The system log can generate one or more of the following events if an unapproved optical transceiver is detected: ■ Unsupported module ■ Optics faulted, incorrectly installed, or not installed ■ Incompatible optics ■ Unknown port SFB status To resolve this issue, do the following: 1. Power off the system. 2. Replace the unapproved optical transceiver with an approved one. 3. Power on the system. **NOTE:** For more information on replacing an optical transceiver, refer to the specific adapter hardware guide available at www.broadcom.com.

5.2 FC Driver Situations and Resolutions

Table 6 lists the FC driver situations and their resolutions.

Table 6: FC Driver Situations and Resolutions

Situation	Resolution
An FC link fails to come up.	If an FC link fails to come up, verify that an 8 Gb/s or 16 Gb/s adapter is not attempting to connect to a 1 Gb/s device. Only 2 Gb/s, 4 Gb/s, and 8 Gb/s devices are supported on 8 Gb/s adapters. Only 4 Gb/s, 8 Gb/s, and 16 Gb/s devices are supported on 16 Gb/s adapters.
If a SAN configuration has 256 targets mapped by the FC driver, any additional added targets do not get a target ID mapping by the driver and cause target discovery to fail.	Removing targets or reinitializing the link does not solve this issue. Unload and reload the driver to reset available target IDs. Ensure that the SAN configuration is correct prior to rebooting the driver. This action clears the driver consistent binding table and frees target IDs for new target nodes.
`rmmod` fails to unload an FC driver module due to `ERROR: Module lpfc is in use.`	This message can appear when you attempt to remove the driver and a Logical Volume Group is dependent on the driver. Driver unload is not supported in SAN-booted environments unless the inbox driver supports SAN boot. Otherwise, use the following steps to resolve this situation: 1. Make the Logical Volume Group unavailable. Type: `lvchange -a n` *`xxxxxxx`* The *`xxxxxxx`* parameter is the Volume Group Name. 2. Stop the Emulex HBA Manager application. 3. Stop Device Mapper.

Table 6: FC Driver Situations and Resolutions (Continued)

Situation	Resolution
`rmmod` hangs and the module reference count is 0.	Due to a small race condition in the kernel, it is possible for an `rmmod` command to hang. Issue the `rmmod -w` command. If this does not help, reboot the computer.

`rmmod` fails to unload the driver due to device or resource busy.	This message occurs when you attempt to remove the driver without first stopping the Emulex HBA Manager application or when FC disks connected to an Emulex adapter are mounted. To resolve this situation:
	1. Stop the Emulex HBA Manager application before attempting to unload the driver. The script is located in the `/usr/sbin/ocmanager` directory. Type:
	`./stop_ocmanager`
	2. Unmount any disks connected to the adapter.
	3. Unload the driver. Type:
	`rmmod lpfc`

An `lspci` shows recent Emulex adapters as unknown.	This situation occurs because of the delay in getting new product IDs into the Linux development cycle. There is no resolution at this time.
Slow targets or extended link faults on the storage side might result in storage being marked offline by the SCSI mid-layer and remaining offline (not recovered) when the link faults are corrected.	This version of the driver should eliminate this issue. However, if you experience offline device issues, increase the SCSI command timeout to a value greater than or equal to 60 seconds. Emulex also provides a script that addresses this issue. To access the `lun_change_state.sh` script, go to www.broadcom.com.
Under certain conditions of an I/O load, some targets cannot complete an I/O issued by a Linux initiator within the default timeout of 30 seconds given by the SCSI mid-layer.	If the situation is not corrected, the initiator-to-target condition deteriorates into abort or recovery storms, leading to I/O failures in the block layer. These types of failures are preceded by a SCSI I/O error of hexadecimal 6000000. Emulex provides a script that addresses this issue. To access the `set_target_timeout.sh` script, go to www.broadcom.com.
The FC driver fails to recognize an adapter and logs `unknown IOCB` messages in the system log during driver load.	The adapter is running outdated firmware. Install the latest firmware on the adapter. **NOTE:** Before performing a firmware update, driver installation is required. For more information on installing the driver, see Section 2.2.1, Installing the Emulex FC Driver Kit.
Loading the FC driver on SLES operating systems reports `unsupported module, tainting kernel` in the system log.	This message is logged by the kernel whenever a module that is not shipped with the kernel is loaded. Ignore this message.
The system panics when it is booted with a failed adapter installed.	Remove the failed adapter and reboot the system.
Driver installation fails.	The `elx_lpfc_install.sh` script fails to install the driver. The install script can fail for the following reasons: ■ A previous version of the driver is installed. Run the `elx_lpfc_install.sh --uninstall` script and then try to install the driver. ■ The current driver is already installed. ■ The RHEL or SLES kernel is not supported. Run a supported RHEL or SLES kernel. ■ Required dependencies are not installed.
The `No module lpfc found for kernel KERNELVERSION` RPM error message appears when upgrading the kernel.	These three situations can be resolved by upgrading the kernel. There are two ways to install the driver into an upgraded kernel. The method that you use depends on whether you are updating the driver. ■ Upgrade the kernel using the same version of the driver. ■ Upgrade the kernel using a new version of the driver. For more information on upgrading the kernel, see Chapter 2, Installing and Uninstalling.
A recently upgraded kernel cannot find the ramdisk. After upgrading the kernel, the kernel cannot find the ramdisk, which halts or panics the system.	
The driver is not loaded after a system reboot after upgrading the kernel.	

Table 6: FC Driver Situations and Resolutions (Continued)

Situation	Resolution
Driver uninstall fails.	The `elx_lpfc_install.sh --uninstall` script fails with an error. Try the following solutions: ■ Uninstall the Emulex HBA Manager application; refer to the *Emulex HBA Manager User Guide* for instructions. ■ Unmount all FC disk drives. ■ Unload the `lpfc` driver. ■ Use ``rpm -e `rpm -qa | grep lpfc` `` to uninstall the lpfc driver RPM.
The `elx_lpfc_install.sh` script exit message appears.	The `elx_lpfc_install.sh` script contains exit messages that can be useful in diagnosing installation issues.
The Emulex driver for Linux does not load in ramdisk for a custom-built kernel.	Emulex does not support custom-built kernels. However, the Emulex installation script attempts to install the driver into a ramdisk that follows the naming scheme used by RHEL or SLES kernels. ■ The RHEL naming scheme for ramdisk images is: `/boot/initrd-`***`KERNELVERSION`***`.img` ■ The SLES naming scheme for ramdisk images is: `/boot/initrd` If a custom-built kernel has a ramdisk image that does not follow the appropriate naming scheme, change the name of the image using the following procedure: 1. Change the name of the ramdisk image to match the SLES naming scheme. 2. Update any file links to the ramdisk image. 3. Edit the boot loader configuration file (for example, `/etc/lilo.conf`, `/etc/yaboot.conf`, `/boot/grub/grub.conf`, `/boot/grub/menu.lst`), find any references to the old ramdisk image name, and replace them with the new name. 4. Reboot the system to verify the changes. 5. Install the Emulex Linux driver kit.
The Linux SCSI subsystem sees only eight LUNs when more are present.	Some SCSI drivers do not scan past eight LUNs when the target reports itself as a SCSI-2 device. To resolve this situation, force a SCSI bus scan with the following command: `/usr/sbin/lpfc/lun_scan` SUSE supplies a `/bin/rescan-scsi-bus.sh` script, which can be changed to scan everything.

5.3 Dynamic Firmware Logging

For firmware-related issues, you may want to refer to firmware logs. Dynamic firmware logging allows you to collect the firmware logs as and when required or when you encounter a firmware issue. You can also enable dynamic firmware logging in the absence of the Emulex HBA Manager application to dynamically generate firmware logs directly from the driver.

To dynamically generate firmware logs, perform the following steps:

1. Enable `lpfc_ras_fwlog_buffsize`.

   ```
   echo 1 > /sys/class/scsi_host/host#/lpfc_ras_fwlog_buffsize
   ```

   `lpfc_ras_fwlog_buffsize` is the firmware logging host buffer size. For more information on `lpfc_ras_fwlog_buffsize`, see Table 3, Dynamic FC Driver Parameters.

2. Mount `debugfs`.

   ```
   mount -t debugfs - /debug
   ```

   Use `debugfs` in the absence of the Emulex HBA Manager application.

3. Collect firmware logs.

   ```
   cat /debug/lpfc/fn#/fw_log
   ```

   where `#` is the port for which you want to generate firmware logs.

5.4 FC Driver Log Messages

The following sections describe retrieving and interpreting FC driver log messages.

5.4.1 Retrieving FC Driver Log Messages

FC error log messages are logged in the `/var/log/messages` file.

An example of an FC message follows:

```
Jul 2 04:23:34 daffy kernel: lpfc 0000:03:06.0: 0:1305 Link Down Event x2f2 received Data: x2f2 x20
x110
```

In this example:

- `lpfc 0000:03:06.0` – Identifies the PCI location of the particular FC hardware port.
- `0:` – Indicates Emulex adapter 0.
- `1305` – Indicates a log message number of 1305.

NOTE:

- If Data: is present in a log message, any information following Data: is intended only for Broadcom® Technical Support or Engineering use.
- If an error message instructs you to perform a firmware update, ensure that the driver is installed first. For more information on installing the driver, see Chapter 2, Installing and Uninstalling.

Table 7 lists the groups and defines the associated number ranges.

Table 7: Message Log Table

LOG Message Verbose Mask Definition	Verbose Bit	Verbose Description
LOG_ELS	0x00000001	ELS events
LOG_DISCOVERY	0x00000002	Link discovery events
LOG_MBOX	0x00000004	Mailbox events
LOG_INIT	0x00000008	Initialization events
LOG_LINK_EVENT	0x00000010	Link events
LOG_IP	0x00000020	IP traffic history
LOG_FCP	0x00000040	FCP traffic history
LOG_NODE	0x00000080	Node table events
LOG_TEMP	0x00000100	Temperature sensor events
LOG_BG	0x00000200	BlockGuard events
LOG_MISC	0x00000400	Miscellaneous events
LOG_SLI	0x00000800	SLI events
LOG_FCP_ERROR	0x00001000	Log errors, not underruns
LOG_LIBDFC	0x00002000	Libdfc events
LOG_VPORT	0x00004000	NPIV events
LOG_SECURITY	0x00008000	Security events

Table 7: Message Log Table (Continued)

LOG Message Verbose Mask Definition	Verbose Bit	Verbose Description
LOG_EVENT	0x00010000	CT,TEMP,DUMP, logging
LOG_FIP	0x00020000	FIP events
LOG_FCP_UNDER	0x00040000	FCP underruns errors
LOG_SCSI_CMD	0x00080000	ALL SCSI commands
LOG_NVME	0x00100000	NVME general events
LOG_NVME_DISC	0x00200000	NVME discovery/connect events
LOG_NVME_ABTS	0x00400000	NVME ABTS events
LOG_NVME_IOERR	0x00800000	NVME I/O Error events
LOG_EDIF	0x01000000	External DIF events
LOG_AUTH	0x02000000	Authentication events
LOG_ALL_MSG	0xFFFFFFFF	LOG all messages

5.4.2 FC Error Log Messages and Their Descriptions

The following table lists the FC error log messages and their descriptions.

Table 8: FC Error Log Messages and Their Descriptions

`0111: Dropping received ELS cmd` The driver decided to drop an `ELS Response` ring entry. Data: (1) `ulpStatus` (2) `ulpWord[4]` (3) `ulpTimeout` Severity: Error Log: Always Action: This error could indicate a software driver or firmware issue. If this issue persists, report these errors to Broadcom Technical Support.
`0113: An FLOGI ELS command <elsCmd> was received from DID <did> in Loop Mode` While in Loop Mode, an unknown or unsupported ELS command was received. Data: None Severity: Error Log: Always Action: Check the device DID.
`0115: Unknown ELS command <elsCmd> received from NPORT <did>` Received an unsupported ELS command from a remote `N_Port`. Data: None Severity: Error Log: Always Action: Check the remote `N_Port` for a potential issue.
`0125: FDISC Failed (x%x). Fabric out of resources` The fabric rejected an `FDISC` because the switch cannot support additional virtual ports. Data: `lsRjtError` Severity: Error Log: Always Action: Reconfigure the switch to support more NPIV logins. If this issue persists, contact Broadcom Technical Support.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

0126: FDISC failed ulpStatus ulpWord4 Data: lsRjtError Severity: Error Log: Always Action: Reconfigure the switch to support more NPIV logins. If this issue persists, contact Broadcom Technical Support.
0127: ELS timeout An ELS IOCB command was posted to a ring and did not complete within ULP timeout seconds. Data: (1) elscmd (2) remote_id (3) ulpcommand (4) ulpIoTag Severity: Error Log: Always Action: If no ELS command is going through the adapter, reboot the system. If the issue persists, contact Broadcom Technical Support.
0133: PLOGI: no memory for reg_login Memory allocation error. Data: (1) nlp_DID (2) nlp_state (3) nlp_flag (4) nlp_rpi Severity: Error Log: LOG_ELS Action: Memory allocation error. Check system resources. Unload unused modules.
0134: PLOGI cannot issue reg_login The ELS PLOGI mailbox command has failed. Data: (1) nlp_DID (2) nlp_state (3) nlp_flag (4) nlp_rpi Severity: Error Log: LOG_ELS Action: Check the port and switch configuration.
0135: cannot format reg_login Could not allocate an RPI or DMA buffer for the mailbox command. Data: (1) nlp_DID (2) nlp_state (3) nlp_flag (4) nlp_rpi Severity: Error Log: LOG_ELS Action: None required.
0136: PLOGI completes to NPort <DID> completion A PLOGI has completed for which there is no NDLP. Data: (1) ulpStatus (2) ulpWord[4] Severity: Error Log: LOG_ELS Action: None required.
0137: No retry ELS command <ELS_CMD> to remote Data: (1) ulpStatus (2) ulpWord[4] Severity: Error Log: LOG_ELS Action: None required.
0138: ELS rsp: Cannot issue reg_login for <DID> REG_LOGIN mailbox command failed. Data: (1) nlp_DID (2) nlp_state (3) nlp_flag (4) nlp_rpi Severity: Error Log: LOG_ELS Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0139: Ignoring ELS cmd tag <ioTag> completion Data` This ELS command was aborted. Data: (1) `ulpStatus` (2) `ulpWord[4]` (3) `ulpTimeout` Severity: Error Log: `LOG_ELS` Action: None required.
`0140: PLOGI Reject: invalid name` Invalid node WWN provided. Data: None Severity: Error Log: `LOG_ELS` Action: None required.
`0141: PLOGI Reject: invalid pname` Invalid port WWN provided. Data: None Severity: Error Log: `LOG_ELS` Action: None required.
`0142: PLOGI RSP: Invalid WWN` The PLOGI sent to the port by a remote port had an invalid WWN. Data: None Severity: Error Log: `LOG_ELS` Action: None required.
`0143: SLI4 Adapter Hardware Error Data: <status0>/<status1>` The HBA has encountered an unrecoverable error. Data: None Severity: Error Log: `LOG_INIT` Action: Use HBACMD to retrieve a dump file.
`0144: Not a valid WCQE code: <Completion Code>` The completion queue handler detected an invalid type. Data: None Severity: Error Log: `LOG_SLI` Action: None required.
`0147: Failed to allocate memory for RSCN event` Memory could not be allocated to send the RSCN event to the management application. Data: None Severity: Error Log: `LOG_ELS` Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0148: Failed to allocate memory for LOGO event` Memory could not be allocated to send the LOGO event to the FC transport. Data: None Severity: Error Log: `LOG_ELS` Action: None required.
`0149: Failed to allocate memory for ELS event` Memory could not be allocated to send the ELS event to the FC transport. Data: None Severity: Error Log: `LOG_ELS` Action: None required.
`0200: CONFIG_LINK bad hba state <hba_state>` A CONFIG_LINK mailbox command completed and the driver was not in the right state. Data: None Severity: Error Log: Always Action: Software driver error. If this issue persists, report these errors to Broadcom Technical Support.
`0203: Devloss timeout on WWPN <address> NPort <nlp_DID>` A remote N_Port that was discovered by the driver disappeared for more than lpfc_devloss_tmo seconds. Data: (1) nlp_flag (2) nlp_state (3) nlp_rpi Severity: Error Log: Always Action: If the device generating this message is not a target to which the HBA is connected, this error will not affect the data integrity of the I/O between the HBA and the attached storage and can be ignored.
`0206: Device discovery completion error` This indicates that an uncorrectable error was encountered during device discovery after a link-up. FC devices will not be accessible if this message is displayed. Data: None Severity: Error Log: Always Action: Reboot the system. If this issue persists, report the error to Broadcom Technical Support. Run with verbose mode enabled for more information.
`0207: Device <DID> (<WWN>) sent invalid service parameters. Ignoring device.` Invalid service parameters were received from DID. This remote port is ignored. Data: DID, WWN Severity: Error Log: Always Action: Verify the remote port's configuration. If this issue persists, report the error to Broadcom Technical Support. Run with verbose mode on for more details.
`2559: Block sgl registration required DMAsize <reqlen> great than a page` The request to post SGL pages does not fit on a page. Data: None Severity: Warning Log: `LOG_INIT` Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0221: FAN timeout` A link-up event was received without the login bit set, so the driver waits E_D_TOV for the fabric to send a FAN. If no FAN if received, a FLOGI will be sent after the timeout. Data: None Severity: Warning Log: `LOG_DISCOVERY verbose` Action: None required. The driver recovers from this condition by issuing a FLOGI to the fabric.
`0222: Initial FLOG/FDISC timeout` The driver sent the initial FLOGI or FDISC to the fabric and never received a response. Data: None Severity: Error Log: Always Action: Check fabric configuration. The driver recovers from this situation and continues with device discovery.
`0223: Timeout while waiting for NameServer login` Login request to the NameServer was not acknowledged within R_A_TOV. Data: None Severity: Error Log: Always Action: Check the fabric configuration. The driver recovers from this and continues with device discovery.
`0224: NameServer Query timeout` Node authentication timeout, node Discovery timeout. A NameServer Query to the fabric or discovery of reported remote N_Ports is not acknowledged within R_A_TOV. Data: (1) fc_ns_retry (2) fc_max_ns_retry Severity: Error Log: Always Action: Check fabric configuration. The driver recovers from this situation and continues with device discovery.
`0227: Node Authentication timeout` The driver has lost track of what N_Ports are being authenticated. Data: None Severity: Error Log: Always Action: None required. The driver should recover from this event.
`0228: CLEAR LA timeout` The driver issued a CLEAR_LA that never completed. Data: None Severity: Error Log: Always Action: None required. The driver should recover from this event.
`0230: Unexpected timeout, hba linkstate <link_state>` Discovery has timed out, and the HBA state is not ready. Data: None Severity: Error Log: `LOG_DISCOVERY` Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0231: RSCN timeout` The driver has lost track of which N_Ports have RSCNs pending. Data: (1) fc_ns_retry (2) lpfc_max_ns_retry Severity: Error Log: Always Action: None required. The driver should recover from this event.
`0233: Nodelist not empty` Driver unloaded or hot plug detected a node still in use. Data: None Severity: Error Log: `LOG_DISCOVERY` Action: None required.
`0237: Pending Link Event during Discovery: State <hba_state>` Received link event during discovery. Causes discovery restart. Data: None Severity: Warning Log: `LOG_DISCOVERY verbose` Action: None required, unless this issue persists. If persistent, check cabling.
`0241: NameServer rsp error` The driver received a NameServer response containing a status error. Data: (1) CommandResponse.bits.CmdRsp (2) ReasonCode (3) Explanation (4) fc_flag Severity: Error Log: `LOG_DISCOVERY verbose` Action: Check the fabric configuration. The driver recovers from this situation and continues with device discovery.
`0246: RegLogin failed` The firmware returned a failure for the specified RegLogin. Data: (1) `Did` (2) `mbxStatus` (3) `hbaState` Severity: Error Log: Always Action: This message indicates that the firmware could not do RegLogin for the specified DID. There might be a limitation on how many nodes an HBA can see.
`0249: Cannot issue Register Fabric login: Err %d\` Could not issue the fabric register login, the error value is unique for each possible failure. Data: None Severity: Error Log: `LOG_ELS` Action: None required.
`0251: NameServer login: no memory` Could not allocate memory for the NDLP structure. Data: None Severity: Error Log: `LOG_ELS` Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

0252: Cannot issue NameServer login Could not issue an ELS PLOGI to the NameServer DID. Data: None Severity: Error Log: LOG_ELS Action: Check the port connection and the switch configuration.
0253: Register VPI: Can’t send mbox\ Could not issue the REG_VPI mailbox command for this vPort. Data: None Severity: Error Log: LOG_MBOX Action: None required.
0254: Register VPI: no memory Could not allocate memory for the REG_VPI mailbox command. Data: None Severity: Error Log: LOG_MBOX Action: None required.
0255: Issue FDISC: no IOCB All of the preallocated IOCBs are in use. Data: None Severity: Error Log: LOG_ELS Action: None required.
0256: Issue FDISC: Cannot send IOCB\ Unable to send the fabric IOCB. Data: None Severity: Error Log: LOG_ELS Action: None required.
0257: GID_FT Query error The GID_FT CT request for the NameServer has failed. Data: None Severity: Error Log: LOG_ELS Action: Check the switch configuration.
0258: Register Fabric login error: The REG_LOGIN for the fabric has failed. Data: None Severity: Error Log: LOG_MBOX Action: Check the port connection and the switch configuration.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0259: No NPIV Fabric support` The switch to which the port is connected does not support NPIV. Data: None Severity: Error Log: `LOG_ELS` Action: Check the switch configuration.
`0260: Register NameServer error:` The REG_LOGIN mailbox command has failed for the NameServer. Data: None Severity: Error Log: `LOG_ELS` Action: Check the switch configuration
`0261: Cannot register NameServer login:` Either a memory allocation issue or an invalid parameter was sent to the REG_LOGIN. Data: None Severity: Error Log: `LOG_ELS` Action: At least one message (0142 0121 0133 0134 0135) should precede this message.
`0262: No NPIV Fabric support` The switch to which the port is connected does not support NPIV. Data: None Severity: Error Log: `LOG_ELS` Action: Check the switch configuration.
`0263: Discovery Mailbox error: state:` Either the driver could not allocate resources or it could not send `sparam_mbox` or `cfglink_mbox`. Data: (1) address of `sparam_mbox` command (2) address of `cfglink_mbox` command Severity: Error Log: `LOG_MBOX` Action: Attempt to unload and reload the driver when it is convenient.
`0264: No NPIV Fabric support` The switch to which the port is connected does not support NPIV. Data: None Severity: Error Log: `LOG_ELS` Action: Check the switch configuration.
`0266: Issue NameServer Req <cmdcode> err <rc> Data: <fc_flag> <fc_rscn_id_cnt>` The driver was unable to send the NameServer CT command. Data: (1) `vports fc_flag` (2) `vports fc_rscn_id_cnt` Severity: Error Log: `LOG_DISCOVERY` Action: Check the port and switch configurations.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

0267: NameServer GFF Rsp <did> Error (<ulpStatus> <un.ulpWord[4]>) Data: <fc_flag> <fc_rscn_id_cnt> The NameServer GFF CT request failed. Data: (1) vports fc_flag (2) vports fc_rscn_id_cnt Severity: Error Log: LOG_DISCOVERY Action: Check the port and switch configurations.
0268: NS cmd <cmdcode> Error (<ulpStatus> <un.ulpWord[4]>) The NameServer CT request failed. Data: None Severity: Error Log: LOG_DISCOVERY Action: Check the port and switch configurations.
0271: Illegal State Transition: node <nlp_DID> event <evt>, state <nlp_state> Data: <nlp_rpi> <nlp_flag> The current node state does not have a handler for this event. Data: (1) nlp_rpi (2) nlp_flag Severity: Error Log: LOG_DISCOVERY Action: Verify that all targets are still visible to the SCSI mid-layer.
0272: Illegal State Transition: node <nlp_DID> event <evt>, state <nlp_state> Data: <nlp_rpi> <nlp_flag> The driver is completing a PLOGI but does not have the rcv_plogi flag set. Data: (1) nlp_rpi (2) nlp_flag Severity: Error Log: LOG_DISCOVERY Action: Verify that all targets are still visible to the SCSI midlayer.
0273: Unexpected discovery timeout, vport State x%x The discovery process has timed out. Data: None Severity: Error Log: LOG_DISCOVERY Action: Verify that all targets are still visible.
0274: lpfc_nlp_put: ndlp:x%pusgmap:x%x refcnt:%d, void *)ndlp, ndlp->nlp_usg_map, atomic_read(&ndlp->kref.refcount) Data: None Severity: Warning Log: LOG_NODE Action: None required.
0275: lpfc_nlp_put: ndlp:x%pusgmap:x%x refcnt:%d, void *)ndlp, ndlp->nlp_usg_map, atomic_read(&ndlp->kref.refcount) A kref_put was called again after the node was already inactive. Data: None Severity: Warning Log: LOG_NODE Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

0276: lpfc_nlp_get: ndlp:x%pusgmap:x%x refcnt:%d, void *)ndlp, ndlp->nlp_usg_map, atomic_read(&ndlp->kref.refcount) A kref_get was attempted on a node that was being released. Data: None Severity: Warning Log: LOG_NODE Action: None required.
0277: lpfc_enable_node: ndlp:x%pusgmap:x%x refcnt:%d, void *)ndlp, ndlp->nlp_usg_map, atomic_read(&ndlp->kref.refcount) Enable node was attempted on a node that has already been marked for memory free. Data: None Severity: Warning Log: LOG_NODE Action: None required.
0278: lpfc_enable_node: ndlp:x%pusgmap:x%x refcnt:%d, void *)ndlp, ndlp->nlp_usg_map, atomic_read(&ndlp->kref.refcount) Enable node was attempted on an inactive node. Data: None Severity: Warning Log: LOG_NODE Action: None required.
0280: lpfc_cleanup_node: ndlp:x%pusgmap:x%x refcnt:%d, void *)ndlp, ndlp->nlp_usg_map, atomic_read(&ndlp->kref.refcount) Node cleanup was attempted on a node that has already been marked for memory free. Data: None Severity: Warning Log: LOG_NODE Action: None required.
0281: lpfc_cleanup_node: ndlp:x%pusgmap:x%x refcnt:%d, void *)ndlp, ndlp->nlp_usg_map, atomic_read(&ndlp->kref.refcount) Node cleanup was called to prepare the node for release. Data: None Severity: Warning Log: LOG_NODE Action: None required.
0282: ldid:x%x ndlp:x%pusgmap:x%x refcnt:%d, ndlp->nlp_DID, (void *)ndlp, lpfc_init.c-ndlp->nlp_usg_map, Driver clean-up has found a node that is still on the node list during driver unload or PCI hot plug removal. Data: None Severity: Error Log: LOG_NODE Action: None required.
06119: Failed to allocate mbox cmd memory Mailbox allocation error. Data: None Severity: Error Log: LOG_INIT Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

0285: Allocated DMA memory size <alloclen> is less than the requested DMA memorysize<reqlen> Memory allocation was truncated. Data: None Severity: Error Log: LOG_INIT Action: None required.
0286: lpfc_nlp_state_cleanup failed to allocate statistical data buffer <nlp_DID> Memory allocation failed for node's statistical data. Data: None Severity: Error Log: LOG_INIT Action: None required.
0287: lpfc_alloc_bucket failed to allocate statistical data buffer DID <nlp_DID> Memory allocation failed for node's statistical data. Data: None Severity: Error Log: LOG_NODE Action: None required.
0289: Issue Register VFI failed: Err <rc> The driver could not register the Virtual Fabric Index for the FCFI. Data: None Severity: Error Log: LOG_ELS Action: Check the switch and port configurations.
0290: The SLI4 DCBX asynchronous event is not handled yet The SLI-4 DCBX asynchronous event is not handled yet. Data: None Severity: Error Log: LOG_SLI Action: None required.
0291: Allocated DMA memory size (x%x) is less than the requested DMA memory size (x%x) The asynchronous DCBX events are not handled in the driver. Data: None Severity: Error Log: LOG_INIT Action: Check the switch configuration.
0293: PM resume failed to start worker thread: error=<error> The PCI resume (hot plug) could not start the worker thread for the driver. Data: None Severity: Error Log: LOG_INIT Action: Unload and reload the driver.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0294: PM resume Failed to enable interrupt` The PCI resume (hot plug) could not get an interrupt vector. Data: None Severity: Error Log: `LOG_INIT` Action: Unload and reload the driver.
`0297: Invalid device group<pci_dev_grp>` While unloading the driver, the driver detected a PCI device that it should not have claimed. Data: None Severity: Error Log: `LOG_INIT` Action: None required.
`0299: Invalid SLI revision <sli_rev>` While processing a host attention or unrecoverable error, the driver detected an invalid SLI revision. Data: None Severity: Error Log: `LOG_INIT` Action: None required.
`0300: LATT: Cannot issue READ_LA: Data:<rc>` The link attention handler could not issue a `READ_LA` mailbox command. Data: None Severity: Error Log: `LOG_MBOX` Action: None required.
`0301: READ_SPARAM: no buffers` The driver attempted to issue a `READ_SPARAM` mailbox command to the adapter, but there were no buffers available. Data: None Severity: Warning Log: `LOG_MBOX verbose` Action: This message indicates one of these two issues: (1) Kernel virtual memory is depleted. Check that the system meets minimum RAM requirements for the adapter. Try closing other applications to free some memory. (2) A possible driver buffer management issue. If this issue persists, report the error to Broadcom Technical Support.
`0302: REG_LOGIN: no buffers` The driver attempted to issue a `REG_LOGIN` mailbox command to the adapter, but no buffers were available. Data: (1) Did, (2) flag Severity: Warning Log: `LOG_MBOX verbose` Action: This message indicates one of these two issues: (1) Kernel virtual memory is depleted. Check that the system meets minimum RAM requirements for the adapter. Try closing other applications to free some memory. (2) A possible driver buffer management issue. If this issue persists, report the error to Broadcom Technical Support.
`0313: Ring <ringno> handler: unexpected Rctl <Rctl> Type <Type> received` The RCTL/Type of a received frame did not match any for the configured masks for the specified ring. Data: None Severity: Warning Log: `LOG_SLI verbose` Action: This error could indicate a software driver, firmware, or hardware issue. Report these errors to Broadcom Technical Support.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0303: Ring <ringno> handler: portRspPut <portRspPut> is bigger then rsp ring <portRspMax>` The port rsp ring put index is larger than the size of the rsp ring. Data: None Severity: Error Log: Always Action: This error could indicate a software driver, firmware, or hardware issue. Report these errors to Broadcom Technical Support.
`0304: Stray mailbox interrupt, mbxCommand <mbxcommand> mbxStatus <mbxstatus>` Received a mailbox completion interrupt and there are no outstanding mailbox commands. Data: None Severity: Error Log: Always Action: This error could indicate a hardware or firmware issue. If this issue persists, report the error to Broadcom Technical Support.
`0306: CONFIG_LINK mbxStatus error <mbxStatus> HBA state <hba_state>` The driver issued a `CONFIG_LINK` mailbox command to the HBA that failed. Data: None Severity: Error Log: Always Action: This error could indicate a firmware or hardware issue. Report these errors to Broadcom Technical Support.
`0310: Mailbox command <mbxcommand> timeout` A mailbox command was posted to the adapter and did not complete within 30 seconds. Data: (1) hba_state (2) sli_flag (3) mbox_active Severity: Error Log: Always Action: This error could indicate a software driver or firmware issue. If no I/O is going through the adapter, reboot the system. If this issue persists, report the error to Broadcom Technical Support.
`0312: Ring <ringno> handler: portRspPut <rspPutInx> is bigger then rsp ring <numRiocb>` The IOCB command rings put pointer is ahead of the get pointer. Data: None Severity: Error Log: LOG_SLI Action: None required.
`0313: Ring <ringno> handler: unexpected Rctl <Rctl> Type <Type> received` The RCTL/Type of a received frame did not match any for the configured masks for the specified ring. Data: None Severity: Warning Log: LOG_SLI verbose Action: This error could indicate a software driver, firmware or hardware issue. Report these errors to Broadcom Technical Support.
`0315: Ring <ringno> issue: portCmdGet <local_getidx> is bigger then cmd ring <max_cmd_idx>` The port command ring get index is greater than the size of the command ring. Data: None Severity: Error Log: Always Action: This error could indicate a software driver, firmware or hardware issue. Report these errors to Broadcom Technical Support.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0317: iotag <ulp_IoTag> is out of range: max iotag <max_iotag> wd0 <wd0>` The IoTag in the completed IOCB is out of range. Data: None Severity: Error Log: Always Action: This error could indicate a software driver, firmware or hardware issue. Report these errors to Broadcom Technical Support.
`0318: Failed to allocate IOTAG. last IOTAG is <last_allocated_iotag>` The driver cannot allocate an IOTAG. Display the last value used. Data: None Severity: Error Log: Always Action: This message indicates the adapter HBA I/O queue is full. Typically this happens when heavy I/O is running on a low-end (3-digit) adapter. Upgrade to a higher-end adapter.
`0319: READ_SPARAM mbxStatus error <mbxStatus> hba state <hba_state>` The driver issued a `READ_SPARAM` mailbox command to the HBA that failed. Data: None Severity: Error Log: Always Action: This error could indicate a firmware or hardware issue. Report these errors to Broadcom Technical Support.
`0320: CLEAR_LA mbxStatus error <mbxStatus> hba state <hba_state>` The driver issued a `CLEAR_LA` mailbox command to the HBA that failed. Data: None Severity: Error Log: Always Action: This error could indicate a firmware or hardware issue. Report these errors to Broadcom Technical Support.
`0322: Ring <ringno> handler: unexpected completion IoTag <IoTag>` The driver could not find a matching command for the completion received on the specified ring. Data: (1) ulpStatus, (2) ulpWord[4], (3) ulpCommand, (4) ulpContext Severity: Warning Log: `LOG_SLI verbose` Action: This error could indicate a software driver or firmware issue. If this issue persists, report these errors to Broadcom Technical Support.
`0323: Unknown Mailbox command <mbxCommand> Cmpl` A unknown mailbox command completed. Data: None Severity: Error Log: Always Action: This error could indicate a software driver, firmware, or hardware issue. Report these errors to Broadcom Technical Support.
`0324: Config port initialization error, mbxCmd <mbxCommand> READ_NVPARM, mbxStatus <mbxStatus>` The `READ_NVPARMS` mailbox command failed during port configuration. Data: None Severity: Error Log: Always Action: This error could indicate a software driver, firmware, or hardware issue. Report these errors to Broadcom Technical Support.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0328: Rsp Ring <ring number> error: IOCB Data:` The firmware has returned an error for this IOCB. Data: (1) <iocb word[0]:iocb word[7]>, (2) <rsp word[0]:rsp[word[7]> Severity: Warning Log: `LOG_SLI` Action: None required.
`0330: IOCB wake NOT set` The completion handler associated with the IOCB was never called. Data: (1) timeout (2) timeleft/jiffies Severity: Error Log: Always Action: This error could indicate a software driver, firmware, or hardware issue. If this issue persists, report the error to Broadcom Technical Support.
`0334: Unknown IOCB command` Received an unknown IOCB command completion. Data: (1) type (2) ulpCommand (3) ulpStatus (4) ulpIoTag (5) ulpContext) Severity: Error Log: Always Action: This error could indicate a software driver or firmware issue. If this issue persists, report these errors to Broadcom Technical Support.
`0335: Unknown IOCB command` Received an unknown IOCB command completion. Data: (1) ulpCommand (2) ulpStatus (3) ulpIoTag (4) ulpContext) Severity: Error Log: Always Action: This error could indicate a software driver or firmware issue. If this issue persists, report these errors to Broadcom Technical Support
`0336: Rsp Ring <ringno> error: IOCB` An IOCB error has occurred on the specified ring. Data: (1) ulpWord[0], (2) ulpWord[1], (3) ulpWord[2], (4) ulpWord[3], (5) ulpWord[4], (6) ulpWord[5], (7) irsp+6, (8) irsp+7 Severity: Warning Log: `LOG_SLI verbose` Action: If this issue persists, check the targets. If the targets are not the problem, report the error to Broadcom Technical Support.
`0340: Adapter temperature is OK now` Adapter temperature has reverted to normal range. Data: Temperature in Celsius Severity: Error Log: `LOG_TEMP verbose` Action: No action needed, informational.
`0341: Ring <ringno> Cannot find buffer for an unsolicited iocb tag <un.ulpWord[3]>` No more preallocated buffers are available to handle unsolicited buffers. Data: None Severity: Error Log: `LOG_SLI` Action: Ensure that this port is not being managed by multiple ports.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0342: Ring <ringno> Cannot find buffer for an unsolicited iocb tag <unsli3.sli3Words>` This is a multiple IOCB unsolicited command, and sufficient buffer space cannot be allocated for it. Data: None Severity: Error Log: `LOG_SLI` Action: None required.
`0343: Ring <ringno> Cannot find buffer for an unsolicited iocb tag <un.ulpWord[3]>` No more preallocated buffers are available to handle unsolicited buffers. Data: None Severity: Error Log: `LOG_SLI` Action: None required.
`0344: Ring <ringno> Cannot find buffer for an unsolicited iocb tag <unsli3.sli3Words[7]>` No more preallocated buffers are available to handle unsolicited buffers. Data: None Severity: Error Log: `LOG_SLI` Action: None required.
`0345: Resetting board due to mailbox timeout iocb. tag 0x%x` A mailbox command failed to complete. The driver is resetting the port. Data: None Severity: Error Log: `LOG_MBOX`, `LOG_SLI` Action: If the mailbox command fails again, set the `lpfc_log_verbose` to `LOG_MBOX` and retry.
`0346: Ring <ring number> handler: unexpected ASYNC_STATUS evt_code <evt code> W0 <hex w0> W1 <hex w1> W2 <hex W2> W3 <hex W3> W4 <hex W4> W5 <hex W5> W6 <hex W6> W7 <hex W7> W8 <hex W8> W9 <hex W9> W10 <hex W10> W11<hex W11>` The HBA received an asynchronous event that was not a temperature event. Data: None Severity: Error Log: `LOG_SLI` Action: None required.
`0347: Adapter is very hot, please take corrective action` Adapter temperature is above normal range. Data: Temperature in Celsius Severity: Error Log: `LOG_TEMP verbose` Action: Shut down and remove the HBA. Contact Broadcom Technical Support.
`0348: NameServer login: node freed` The enable mode failed to free up the NameServer login. Data: None Severity: Error Log: `LOG_ELSI` Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

0349: rc should be MBX_SUCCESS The next mailbox command on the mailbox queue has failed. Data: None Severity: Error Log: LOG_MBOX, LOG_SLI Action: None required.
0350: rc should have been MBX_BUSY Attempting to unregister a default RPI from an interrupt context, and the mailbox state is not busy. Data: None Severity: Error Log: LOG_MBOX, LOG_SLI Action: None required.
0351: Config MSI mailbox command failed, mbxCmd <u.mb.mbxComm>, mbxStatus <u.mb.mbxStatus> The mailbox command sent to the firmware to configure the adapter to use MSI-X has failed. Data: None Severity: Warning Log: LOG_MBOX Action: Ensure that the hardware platform supports MSI-X.
0352: Config MSI mailbox command failed, mbxCmd <u.mb.mbxCommand>, mbxStatus <u.mb.mbxStatus> The mailbox command sent to the firmware to configure the HBA to use MSI-X has failed. Data: None Severity: Error Log: LOG_MBOX Action: Ensure that the hardware platform supports MSI-X.
0353: Active Mailbox cleared - mailbox timeout exiting The mailbox timeout handler has determined that the driver is in the process of completing this mailbox command. Data: None Severity: Error Log: LOG_MBOX, LOG_SLI Action: None required.
0358: MSI-X interrupt with no EQE SLI-4 adapter interrupt on the fast path but there is no associated EQE. Data: None Severity: Warning Log: LOG_SLI Action: None required.
0360:Unsupported EQ count. <entry_count> Cannot create an event queue of this size. Data: None Severity: Error Log: LOG_SLI Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0361:Unsupported CQ count. <entry_count>` Cannot create an completion queue of this size. Data: None Severity: Error Log: LOG_SLI Action: None required.
`0362:Unsupported MQ count. <entry_count>` Cannot create MQ of this size. Data: None Severity: Error Log: LOG_SLI Action: None required.
`0364:Invalid param:` SLI-4: The post-SGL function was passed an invalid XRI. Data: None Severity: Error Log: LOG_SLI Action: None required.
`0365:Slow-path CQ identifier <CQID> does not exist:` The completion queue identifier passed in the event queue entry does not reference a valid completion queue. Data: None Severity: Error Log: LOG_SLI Action: None required.
`0366:Not a valid completion event: majorcode=<major code hex>, minor-code=<minor code hex>` The major or minor code in the Event Queue field is invalid. Data: None Severity: Error Log: LOG_SLI Action: None required.
`0368: Mis-matched fast-path completion queue identifier: eqcqid=%d, fcpcqid=%d` The completion queue identifier in the event queue entry does not match the `fcp_cqid` that was passed into the routine. Data: None Severity: Error Log: LOG_SLI Action: None required.
`0369: No entry from completion queue qid` No completions in the completion queue were referenced by `fcp_cqid`. Data: None Severity: Error Log: LOG_SLI Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0370: Invalid completion queue type <type>` The event queue entry is not for a mailbox or a work queue entry. Data: None Severity: Error Log: LOG_SLI Action: None required.
`0372: iotag <iotag> lookup error: max iotag (<sli.last_iotag>) iocb_flag <iocb_flag>` The IOCB lookup cannot be performed because the `iotag` is out of range. Data: None Severity: Error Log: LOG_SLI Action: None required.
`0373: FCP complete error: status=<status> hw_status=<hw status>, total_data_specified=<total data transferred>, parameter=<rsp word[4]>, word3=<wcqe word 3>` Logs the FCP failure. Status and parameter are equivalent to ulpStatus and ulpWord[4]. Data: None Severity: Warning Log: LOG_SLI Action: None required.
`0374: FCP complete with no corresponding cmdiocb: iotag <iocb iotag>` No IOCB was on the in-progress list that matched this iotag. Data: None Severity: Warning Log: LOG_SLI Action: None required.
`0375: FCP cmdiocb not callback function iotag: <iocb iotag>` The IOCB found for this iotag does not have a completion handler set in it. Data: None Severity: Warning Log: LOG_SLI Action: None required.
`0377: Error <rc> parsing vpd. Using defaults.` Could not parse the VPD data, so the driver is using the default values. Data: None Severity: Error Log: Always Action: None required.
`0378: No support for fcpi mode.` Could not configure the port to run in FCP initiator mode. Data: None Severity: Warning Log: LOG_MBOX, LOG_SLI Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0379: Feature Mismatch Data: <req ftr word2 hex> <req_ftr word3 hex> <cfg_enable_npiv> <max vpi hex>` The features passed in to the driver as module parameters do not match what the firmware can do. Setting to default values. Data: None Severity: Warning Log: LOG_MBOX, LOG_SLI Action: None required.
`0381: Error %d during queue setup.` Could not set up all the queues that the driver requires to exchange I/Os with the HBA. Data: None Severity: Error Log: LOG_MBOX, LOG_SLI Action: Reload the driver.
`0382: READ_SPARAM command failed status <issue status>, mbxStatus <mailbox status>` The `READ_SPARAM` mailbox command has failed during initialization. The HBA has been set to error state. Data: None Severity: Error Log: LOG_MBOX, LOG_SLI: Action: Perform a dump with HBACMD and then try reloading the driver.
`0384: There is pending active mailbox cmd` The mailbox commands have overlapped. This command should have been added to the mailbox queue. Data: None Severity: Error Log: LOG_MBOX, LOG_SLI Action: None required.
`0385: rc should have been MBX_BUSY` The completion handler for `REG_LOGIN` detected the `IMMED_UNREG` flag and tried to issue the `UNREG_LOGIN` command from an interrupt level. The mailbox status should still be busy. Data: None Severity: Error Log: LOG_MBOX, LOG_SLI Action: None required.
`0386: ELS complete with no corresponding cmdiocb: iotag <iotag>` The completion that the ISR is handling cannot find a tag associated with the IOTAG. Data: None Severity: Warning Log: LOG_SLI Action: None required.
`0387:NO IOCBQ data: txq_cnt=<txq_cnt> iocb_cnt=<iocb_cnt> els_txcmplq_cnt=<els_txcmplq_cnt>` Failed to get an IOCBQ from the list of available IOCBQs. Data: None Severity: Error Log: LOG_SLI Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0388:Not a valid WCQE code: x<hex cqe_code>` The event code is invalid. This event will be dropped. Data: None Severity: Error Log: LOG_SLI Action: Ensure that the adapter firmware is current.
`0391:Error during rpi post operation` The driver was trying to post pages to the firmware to keep target login information and encountered a failure. Data: None Severity: Error Log: LOG_MBOX, LOG_SLI Action: Unload and reload the driver.
`0393:Error <rc> during rpi post operation` The driver was trying to post pages to the firmware to keep target login information and encountered a failure. Data: None Severity: Error Log: LOG_MBOX, LOG_SLI Action: Unload and reload the driver.
`0395: The mboxq allocation failed` The asynchronous link event handler could not allocate a mailbox command to issue the `READ_LA` (read link attention) mailbox command. Data: None Severity: Error Log: LOG_SLI Action: None required.
`0396:The lpfc_dmabuf allocation failed` The asynchronous link event handler could not allocate a mailbox command to issue the `READ_LA` mailbox command. Data: None Severity: Error Log: LOG_SLI Action: None required.
`0397:The mbuf allocation failed` The asynchronous link event handler could not allocate DMA for the `READ_LA` mailbox command. Data: None Severity: Error Log: LOG_SLI Action: None required.
`0398 Invalid link fault code: <hex link_fault>` The attempt to read the link attention register returned an unknown value. Data: None Severity: Error Log: LOG_INIT Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

0399 Invalid link attention type: <hex link_type> The READ_LA mailbox command returned an invalid link type. Data: None Severity: Error Log: LOG_INIT Action: None required.
0401: Ignoring change to nodev_tmo because devloss_tmo is set Attempting to change the nodev timeout when the devloss has already been set. Data: None Severity: Error Log: LOG_INIT Action: None required.
0402:Cannot find virtual addr for buffer tag on ring <ringno> A DMA buffer is unavailable for this unsolicited command. Data: (1) tag (2) next (3) prev (4) postbufq_cnt Severity: Error Log: LOG_INIT Action: None required.
0404: lpfc_devloss_tmo attribute cannot be set to <val>, allowed range is [<LPFC_MIN_DEVLOSS_TMO>, <LPFC_MAX_DEVLOSS_TMO>] Attempt to set the devloss timeout value is outside the allowed range. Data: None Severity: Error Log: LOG_INIT Action: Set the devloss timeout between the minimum and maximum devloss range.
0405: lpfc_link_speed attribute cannot be set to %d, allowed values are ["LPFC_LINK_SPEED_STRING"] Attempt to set the link speed value is outside the allowed range. Data: None Severity: Error Log: LOG_INIT Action: Set the link speed between 0 and the maximum.
0406: Adapter maximum temperature exceeded <temperature>, taking this port offline The driver has received an error for the HBA indicating that the maximum allowable temperature has been exceeded. Data: (1) work_hs (2) work_status[0] (3) work_status[1] Severity: Error Log: LOG_INIT Action: Ensure that the server fans are not blocked. Shut down the server if the airflow is restricted.
0407: Ignoring nodev_tmo module parameter because devloss_tmo is set. Both module parameters (nodev and devloss) were set so the driver is ignoring the nodev parameter. Data: None Severity: Error Log: LOG_INIT Action: Set only one of these parameters.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0410: Cannot find virtual addr for mapped buf on ring <ringno>` The driver cannot find the specified buffer in its mapping table. Therefore, it cannot find the virtual address needed to access the data. Data: (1) phys (2) next (3) prev (4) postbufq_cnt Severity: Error Log: Always Action: This error could indicate a software driver or firmware issue. If this issue persists, report these errors to Broadcom Technical Support.
`0421: MSI-X slow-path request_irq failed <rc>` The kernel API to request an IRQ has failed. Data: None Severity: Warning Log: LOG_INIT Action: Use module parameter `lpfc_use_msi=0 (INTx).`
`0422: lpfc_restrict_login attribute cannot be set to <val>, allowed range is [0, 1]` Attempt to set the restrict login parameter to something other than on or off. Data: None Severity: Error Log: LOG_INIT Action: Use 0 (Off) or 1 (On)
`0423: lpfc_"#attr" attribute cannot be set to %d, allowed range is ["#minval", "#maxval"]` This compile time macro is used by several module parameters during initialization. Each module parameter has its own minimum and maximum values that are displayed. Data: None Severity: Error Log: LOG_INIT Action: Set the module parameter between the minimum and maximum values.
`0424: lpfc_"#attr" attribute cannot be set to %d, allowed range is ["#minval", "#maxval"]` This is a compile time macro that is used by several module parameters to set the value. Data: None Severity: Error Log: LOG_INIT Action: Set the module parameter between the minimum and maximum values.
`0425:lpfc_restrict_login attribute cannot be set to %d, allowed range is [0, 1]` The module parameter `lpfc_restrict_login` can only be set to 0 (off) or 1 (on). Data: None Severity: Error Log: LOG_INIT Action: Set `lpfc_restrict_login = [0,1].`
`0426: Failed to enable interrupt` The driver failed to start the interrupt. Data: None Severity: Error Log: LOG_INIT Action: Unload and reload the driver.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0427: Cannot re-enable interrupt after slot reset` The driver was unable to enable the interrupt after an HBA reset. Data: None Severity: Error Log: LOG_INIT Action: Unload and reload the driver.
`0429: MSI-X fast-path request_irq failed (<rc>)` The driver received an error for the `request_irq_call`. Data: None Severity: Warning Log: LOG_INIT Action: Unload and reload the driver.
`0430: PM resume Failed to enable interrupt` The driver power management resume function could not enable the interrupt. Data: None Severity: Error Log: LOG_INIT Action: Perform another PM suspend and resume or HBA reset.
`0431: Failed to enable interrupt.` The driver failed to start the interrupt. Data: None Severity: Error Log: LOG_INIT Action: Unload and reload the driver.
`0433: Wakeup on signal: rc=<rc>` A signal other than the LPFC_DATA_READY was received on the worker thread. Data: None Severity: Error Log: LOG_ELS Action: Unload and reload the driver.
`0434: PM resume failed to start worker thread: error=<error>` The driver's power management resume function could not start the worker thread. Data: None Severity: Error Log: LOG_INIT Action: Unload and reload the driver.
`0435: Adapter failed to get Option ROM version status <rc>` The driver could not read the HBA option ROM. Data: None Severity: Error Log: LOG_INIT Action: Reset the HBA. Ensure the adapter firmware is current.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0436: Adapter failed to init, timeout, status reg <status>` The adapter failed during powerup diagnostics after it was reset. Data: None Severity: Error Log: Always Action: This error could indicate a hardware or firmware issue. If this issue persists, report the error to Broadcom Technical Support.
`0437: Adapter failed to init, chipset, status reg <status>` The adapter failed during powerup diagnostics after it was reset. Data: None Severity: Error Log: Always Action: This error could indicate a hardware or firmware issue. If this issue persists, report the error to Broadcom Technical Support.
`0438: Adapter failed to init, chipset, status reg <status>` The adapter failed during powerup diagnostics after it was reset. Data: None Severity: Error Log: Always Action: This error could indicate a hardware or firmware issue. If this issue persists, report the error to Broadcom Technical Support.
`0439: Adapter failed to init, mbxCmd <mbxCommand> READ_REV, mbxStatus <mbxStatus>` Adapter initialization failed when issuing a `READ_REV` mailbox command. Data: None Severity: Error Log: Always Action: This error could indicate a hardware or firmware issue. If this issue persists, report the error to Broadcom Technical Support.
`0440: Adapter failed to init, READ_REV has missing revision information` A firmware revision initialization error was detected. Data: None Severity: Error Log: Always Action: This error could indicate a hardware or firmware issue. Install the latest firmware revision. If this issue persists, report the error to Broadcom Technical Support.
`0442: Adapter failed to init, mbxCmd <mbxCommand> CONFIG_PORT, mbxStatus <mbxStatus>` Adapter initialization failed when issuing a `CONFIG_PORT` mailbox command. Data: (1) hbainit Severity: Error Log: Always Action: This error could indicate a hardware or firmware issue. If this issue persists, report the error to Broadcom Technical Support.
`0446: Adapter failed to init, mbxCmd <mbxCommand> CFG_RING, mbxStatus <mbxStatus>, ring <num>` Adapter initialization failed when issuing a `CFG_RING` mailbox command. Data: None Severity: Error Log: Always Action: This error could indicate a hardware or firmware issue. If this issue persists, report the error to Broadcom Technical Support.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0448: Adapter failed to init, mbxCmd <mbxCommand> READ_SPARM, mbxStatus <mbxStatus>` Adapter initialization failed when issuing a `READ_SPARM` mailbox command. Data: None Severity: Error Log: Always Action: This error could indicate a hardware or firmware issue. If this issue persists, report the error to Broadcom Technical Support.
`0449: lpfc_%attr attribute cannot be initialized to %d, allowed range is [%min, %max]` The `sysfs` attribute value written exceeds attribute range. Data: (1) attribute name (2) value written (3) minimum value (3) maximum value Severity: Error Log: Always Action: Write a value within the supported range.
`0450: lpfc_%attr attribute cannot be set to %d, allowed range is [%min, %max]` The `sysfs` attribute value written exceeds attribute range. Data: (1) attribute name (2) value written (3) minimum value (3) maximum value Severity: Error Log: Always Action: Write a value within the supported range.
`0453: Adapter failed to init, mbxCmd <mbxCommand> READ_CONFIG, mbxStatus <mbxStatus>` Adapter initialization failed when issuing a `READ_CONFIG` mailbox command. Data: None Severity: Error Log: Always Action: This error could indicate a hardware or firmware issue. If this issue persists, report the error to Broadcom Technical Support.
`0456: Adapter failed to issue ASYNCEVT_ENABLE mbox status x%x` The mailbox command to enable an asynchronous event notification failed. Data: None Severity: Error Log: LOG_INIT Action: Ensure the adapter firmware is current. Reload the driver.
`0457: Adapter Hardware Error` The driver received an interrupt indicating a possible hardware issue. Data: (1) status (2) status1 (3) status2 Severity: Error Log: Always Action: This error could indicate a hardware or firmware issue. If this issue persists, report the error to Broadcom Technical Support.
`0458: Bring adapter online` The FC driver received a request to bring the adapter online. This issue might occur when running `lputil`. Data: None Severity: Warning Log: LOG_INIT verbose Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

0459: Adapter heartbeat still outstanding:last compl time was <time> ms. The Heartbeat mailbox command failed. Data: None Severity: Error Log: LOG_INIT Action: Ensure that the adapter firmware is current. Reload the driver.
0460: Bring adapter offline The FC driver received a request to bring the adapter offline. This issue might occur when running lputil. Data: None Severity: Warning Log: LOG_INIT verbose Action: None required.
0466: Outstanding I/O when waiting for I/O to drain The I/O is still pending while attempting to stop the driver. Data: None Severity: Warning Log: LOG_INIT Action: None required.
0467: lpfc_topology attribute cannot be set to %d, allowed range is [0, 6], phba->brd_no, val. The lpfc_topology module parameter is invalid. Data: None Severity: Error Log: LOG_INIT Action: Use a value in the valid range.
0468: lpfc_restrict_login must be 0 for Physical ports. Cannot restrict the login for the physical port. Data: None Severity: Error Log: LOG_INIT Action: None required.
0469: lpfc_link_speed attribute cannot be set to <val>, allowed values are [list of val] The link speed module parameter is invalid. Data: None Severity: Error Log: LOG_INIT Action: Use a link speed parameter in the valid range.
0472: Unknown PCI error state: <state> The PCI bus detected an error. Data: None Severity: Error Log: LOG_INIT Action: Issue an HBA reset.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0474: Unable to allocate memory for issuing MBOX_CONFIG_MSI command` Mailbox memory pool allocation error. Data: None Severity: Error Log: LOG_INIT Action: None required.
`0475: Not configured for supporting MSI-X cfg_use_msi: 0x%x` The `lpfc_use_msi` module parameter should have been set to 2. Data: None Severity: Error Log: LOG_INIT Action: Set module parameter `lpfc_use_msi= 2`.
`0476: HBA not supporting SLI-3 or later SLI Revision: <sli_rev>` The HBA does not support SLI-3 or SLI-4. Data: None Severity: Error Log: LOG_INIT Action: This HBA does not support msi. Set `lpfc_use_msi = 0`.
`0478: MSI request_irq failed (<rc>).` The `request_irq` kernel API failed. Data: None Severity: Warning Log: LOG_INIT Action: Set `lpfc_use_msi = 0`.
`0479: Deferred Adapter Hardware Error` An adapter hardware error was sent to the driver. Data: (1) work_hs, (2) work_status[0], (3) work_status[1] Severity: Error Log: LOG_INIT Action: Perform a dump using HBACMD.
`0486: MSI-X fast-path (<index>) request_irq failed (<rc>).` The `request_irq` kernel API failed. Data: None Severity: Warning Log: LOG_INIT Action: Set module parameter `lpfc_use_msi = 0`.
`0490: MSI request_irq failed (<rc>).` The `request_irq` kernel API failed. Data: None Severity: Warning Log: LOG_INIT Action: Set module parameter l`pfc_use_msi = 0`.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0492: Unable to allocate memory for issuing SLI_CONFIG_SPECIAL mailbox command` Mailbox memory pool allocation error. Data: None Severity: Error Log: LOG_INIT Action: None required.
`0493: SLI_CONFIG_SPECIAL mailbox failed with status<rc>` Mailbox command failed. Data: None Severity: Error Log: LOG_INIT Action: Ensure that the adapter firmware is current. Unload and reload the driver.
`0494: Unable to allocate memory for issuing "SLI_FUNCTION_RESET mailbox command"` Mailbox memory pool allocation error. Data: None Severity: Error Log: LOG_INIT Action: None required.
`0495: SLI_FUNCTION_RESET mailbox failed with status <shdr_status> add_status <shdr_add_status>, mbx status <rc>.` Mailbox command failed. Data: None Severity: Error Log: LOG_INIT Action: Reset the HBA.
`0497: Failed allocate fast-path IO WQ (<wq index>)` The event queue for the fast path was not allocated. Data: None Severity: Error Log: LOG_INIT Action: Unload and reload the driver.
`0499: Failed allocate fast-path FCP CQ (<fcp_cqidx>).` The completion queue event for the fast path could not be allocated. Data: None Severity: Error Log: LOG_INIT Action: Unload and reload the driver.
`0500: Failed allocate slow-path mailbox CQ` Failed to allocate slow-path mailbox CQ. Data: None Severity: Error Log: LOG_INIT Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0501: Failed allocate slow-path ELS CQ` Failed to allocate slow-path ELS CQ. Data: None Severity: Error Log: LOG_INIT Action: None required.
`0503: Failed allocate fast-path FCP` Failed to allocate fast-path FCP. Data: None Severity: Error Log: LOG_INIT Action: None required.
`0504: Failed allocate slow-path ELS WQ` Failed to allocate slow-path ELS WQ. Data: None Severity: Error Log: LOG_INIT Action: None required.
`0505: Failed allocate slow-path ELS MQ` Data: None Severity: Error Log: LOG_INIT Action: None required.
`0506: Failed allocate receive HRQ\n` Data: None Severity: Error Log: LOG_INIT Action: None required.
`0507: Failed allocate receive DRQ` Failed to allocate receive DRQ. Data: None Severity: Error Log: LOG_INIT Action: None required.
`0523: Failed setup of fast-path EQ <fcp_eqidx>, rc = <rc>` The fast-path EQ setup failed. Data: None Severity: Error Log: LOG_INIT Action: None required.
`0528: Mailbox CQ not allocated` The mailbox CQ is not allocated. Data: None Severity: Error Log: LOG_INIT Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0530: ELS CQ not allocated` The ELS CQ is not allocated. Data: None Severity: Error Log: LOG_INIT Action: None required.
`0535: Failed setup of fast-path FCP WQ <fcp_wqidx>, rc = <rc>` The fast-path FCP WQ setup failed. Data: None Severity: Error Log: LOG_INIT Action: None required.
`0540: Receive Queue not allocated` The Receive Queue is not allocated. Data: None Severity: Error Log: LOG_INIT Action: None required.
`0542: lpfc_create_static_vport failed to allocate mailbox memory` Failed to allocate mailbox memory for vPort creation. Data: None Severity: Error Log: LOG_INIT Action: None required.
`0543: lpfc_create_static_vport failed to allocate vport_info\n"))` Failed to allocate `vport_info`. Data: None Severity: Error Log: LOG_INIT Action: None required.
`0544: lpfc_create_static_vport failed to issue dump mailbox command ret <rc> status <mbxStatus>` Failed to issue a dump mailbox command for static vPort creation. Data: None Severity: Warning Log: LOG_INIT Action: None required.
`0545: lpfc_create_static_vport bad information header 0x%x 0x%x\n"\, le32_to_cpu(vport_info->signature), le32_to_cpu(vport_info->rev) & VPORT_INFO_REV_MASK);` Invalid information header; the signature or revision is invalid. Data: None Severity: Error Log: LOG_INIT Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0546: lpfc_create_static_vport failed to create vport` Failed to create a vPort. Data: None Severity: Warning Log: LOG_INIT Action: None required.
`0582: Error <rc> during sgl post operation` The SGL post operation failed. Data: None Severity: Error Log: LOG_MBOX, LOG_IP verbose Action: None required.
`0602: Failed to allocate CQ_EVENT entry` Failed to allocate a CQ_EVENT entry. Data: None Severity: Error Log: LOG_ SLI Action: None required.
`0603: Invalid work queue CQE subtype (x%x)\n", cq-<subtype>` Invalid work queue CQE. Data: None Severity: Error Log: LOG_ SLI Action: None required.
`0700: Bus Reset on target <i> failed` The bus reset for the specified target failed. Data: None Severity: Error Log: LOG_ FCP Action: None required.
`0704: At limitation of <total> preallocated command buffers` The maximum number of command buffers have already been allocated. Data: None Severity: Warning Log: LOG_FCP verbose Action: None required.
`0705: Allocation request of <num> command buffers will exceed max of <hba_queue_depth>. Reducing allocation request to <size>` The number of command buffers requested will exceed the maximum so a smaller quantity will be allocated. Data: None Severity: Warning Log: LOG_FCP verbose Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0708: Allocation request of <num_to_alloc> command buffers did not succeed. Allocated <num_allocated> buffers.` The allocation request for the specified command buffers did not succeed. However, the specified number of buffers are allocated. Data: None Severity: Warning Log: LOG_FCP Action: None required.
`0713: SCSI layer issued Device Reset (%d, %d)` A device reset was issued. Data: None Severity: Error Log: LOG_FCP Action: None required.
`0714: SCSI layer issued bus reset` The SCSI layer is requesting the driver to abort all I/Os to all targets on this HBA. Data: (1) ret Severity: Error Log: Always Action: Check the state of the targets in question.
`0721: Device Reset rport failure: rdata <rdata>` The reset of the R_Port failed. Data: None Severity: Error Log: LOG_FCP Action: None required.
`0722: Target Reset rport failure: rdata <rdata>` The reset of the target failed. Data: None Severity: Error Log: LOG_FCP Action: None required.
`0723: SCSI layer issued Target Reset (%d, %d)` The SCSI layer issued a target reset. Data: None Severity: Error Log: LOG_FCP Action: None required.
`0724: I/O flush failure for context <“LUN”,”TGT”,”HOST”,”Unknown”>: cnt <cnt>` The I/O flush to the LUN, target, or host failed. Data: None Severity: Error Log: LOG_FCP Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`0727: TMF <cmd> to TGT <TGT#> LUN <LUN#> failed (<ulpStatus>, <ulpWord[4]>)` The task management command failed. Data: None Severity: Error Log: LOG_FCP Action: None required.
`0748: Abort handler timed out waiting for abort to complete:ret <status> D <target id> LUN <lun id>` The abort handler timed out waiting for abort to complete. Data: None Severity: Error Log: Always Action: None required.
`0749: SCSI layer issued abort device` The SCSI layer aborted a device. Data: (1) ret, (2) id, (3) lun, (4) snum Severity: Warning Log: LOG_FCP verbose Action: None required.
`0915 Register VPI failed:<mbxStatus>` Could not register the VPI. Data: None Severity: Error Log: LOG_MBOX Action: None required.
`1268 - Failed to allocate ndlp for oxid:x%x SID:x%x` Failed to allocate NDLP. Data: None Severity: Error Log: LOG_ELS Action: None required.
`1302: Invalid speed for this board: Reset link speed to auto: <cfg_link_speed>` The driver is reinitializing the link speed to autodetect. Data: None Severity: Warning Log: LOG_LINK_EVENT verbose Action: None required.
`1303: Link Up Event <eventTag> received` A link-up event was received. It is possible for multiple link events to be received together. Data: (1) fc_eventTag (2) granted_AL_PA (3) UlnkSpeed (4) alpa_map[0] Detail: If link events received, log (1) last event number received, (2) AL_PA granted, (3) Link speed, (4) number of entries in the loop init LILP AL_PA map. An AL_PA map message is also recorded if LINK_EVENT verbose mode is set. Each AL_PA map message contains 16 AL_PAs. Severity: Error Log: Always Action: If numerous link events are occurring, check the physical connections to the FC network.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

1304: Link Up Event ALPA map A link-up event was received. Data: (1) wd1, (2) wd2, (3) wd3, (4) wd4 Severity: Warning Log: LOG_LINK_EVENT verbose Action: If numerous link events are occurring, check the physical connections to the FC network.
1305: Link Down Event <eventTag> received A link-down event was received. Data: (1) fc_eventTag (2) hba_state (3) fc_flag Severity: Error Log: Always Action: If numerous link events are occurring, check the physical connections to the FC network.
1306: Link Up Event in loop back mode x%x received Data: x%x x%x x%x x%x Link-up notification; configured for loopback. Data: (1) fc_eventTag (2) granted_AL_PA (3) UlnkSpeed (4) alpa_map[0] Severity: Error Log: LOG_LINK_EVENT Action: None required.
1308- Link Down Event in loop back mode x%x received Data: x%x x%x x%x Link Down Event received in loop back mode. Data: eventTag, fc_eventTag, port_state, fc_flag Severity: Error LOG: LOG_LINK_EVENT Action: None required.
1309: Link Up Event npiv not supported in loop topology NPIV is not supported in loop topology. Data: None Severity: Error Log: LOG_LINK_EVENT Action: None required.
1310: Menlo Maint Mode Link up Event <eventTag> rcvd The link is up in maintenance mode; only management commands are allowed. Data: (1) fc_eventTag (2) port_state (3) vport fc_flag Severity: Error Log: LOG_LINK_EVENT Action: None required.
1312: Link Down Event <eventTag> received Maintenance mode link-up notification received without entering link-down. Data: (1) fc_eventTag (2) port_state (3) vport fc_flag Severity: Error Log: LOG_LINK_EVENT Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`1400 - Failed to update xri-sgl size and mapping: %d` Updatation of host els xri-sgl sizes and mappings failed. Data: None Severity: Error LOG: LOG_MBOX | LOG_SLI Action: None required.
`1401: Failed to enable pci device.` Failed to enable PCI device during initialization. Data: None Severity: Error Log: LOG_INIT Action: None required.
`1402: Failed to set up pci memory space.` PCI initialization failed. Data: None Severity: Error Log: LOG_INIT Action: None required.
`1404: Failed to set up driver resource.` Driver resource initialization failed. Data: None Severity: Error Log: LOG_INIT Action: None required.
`1405: Failed to initialize iocb list.` Driver resource initialization failed. Data: None Severity: Error Log: LOG_INIT Action: None required.
`1406: Failed to set up driver resource.` Initialization failed to set up driver resource. Data: None Severity: Error Log: LOG_INIT Action: None required.
`1407: Failed to create scsi host.` Initialization failed to create SCSI host. Data: None Severity: Error Log: LOG_INIT Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`1408: Failure HBA POST Status: sta_reg=0x%x, ""perr=x%x, sfi=x%x, nip=x%x, ipc=x%x, xrom=x%x, ""dl=x%x, pstatus=x%x\n", sta_reg.word0, bf_get(lpfc_hst_state_perr, &sta_reg),` The HBA POST failed. Data: None Severity: Error Log: LOG_INIT Action: None required.
`1410: Failed to set up pci memory space.` Initialization failed to set up PCI memory space. Data: None Severity: Error Log: LOG_INIT Action: None required.
`1412: Failed to set up driver resource.` Initialization failed to set up driver resource. Data: None Severity: Error Log: LOG_INIT Action: None required.
`1413: Failed to initialize iocb list.` Initialization failed to initialize the IOCB list. Data: None Severity: Error Log: LOG_INIT Action: None required.
`1414: Failed to set up driver resource.` Initialization failed to set up driver resource. Data: None Severity: Error Log: LOG_INIT Action: None required.
`1415: Failed to create scsi host.` Initialization failed to create SCSI host. Data: None Severity: Error Log: LOG_INIT Action: None required.
`1416: Failed to allocate sysfs attr` Initialization failed to `sysfs` attribute. Data: None Severity: Error Log: LOG_INIT Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`1418: Invalid HBA PCI device group: <dev_grp>` Invalid HBA PCI device group detected. Data: None Severity: Error Log: LOG_INIT Action: None required.
`1419: Invalid HBA PCI device group: <dev_grp>` Invalid HBA PCI device group detected. Data: None Severity: Error Log: LOG_INIT Action: None required.
`1420: Invalid HBA PCI device group:<dev_grp>` Invalid HBA PCI device group detected. Data: None Severity: Error Log: LOG_INIT Action: None required.
`1421: Failed to set up hba` Initialization failed to set up the HBA. Data: None Severity: Error Log: LOG_INIT Action: None required.
`1422: HBA Unrecoverable error: uerr_lo_reg=<ue lo>, uerr_hi_reg=<ue hi>, online0_reg=<Online0>, online1_reg=<Online1>` The HBA notified the driver that it encountered an unrecoverable error. Data: None Severity: Error Log: LOG_INIT Action: Perform a dump from the Emulex HBA Manager application. Then, unload and reload the driver.
`1423: HBA Unrecoverable error: uerr_lo_reg=<ue lo>, uerr_hi_reg=<ue hi>, online0_reg=<Online0>, online1_reg=<Online1>` The HBA notified the driver that it encountered an unrecoverable error. Data: None Severity: Error Log: LOG_INIT Action: Perform a dump from the Emulex HBA Manager application. Then, unload and reload the driver.
`1424: Invalid PCI device group:<pci_dev_grp>` Invalid HBA PCI device group detected. Data: None Severity: Error Log: LOG_INIT Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`1425: Invalid PCI device group: <pci_dev_grp>` Invalid HBA PCI device group detected. Data: None Severity: Error Log: LOG_INIT Action: None required.
`1426: Invalid PCI device group: <pci_dev_grp>` Invalid HBA PCI device group detected. Data: None Severity: Error Log: LOG_INIT Action: None required.
`1427: Invalid PCI device group: <pci_dev_grp>` Invalid HBA PCI device group detected. Data: None Severity: Error Log: LOG_INIT Action: None required.
`1428: Invalid PCI device group: <pci_dev_grp>` Invalid HBA PCI device group detected. Data: None Severity: Error Log: LOG_INIT Action: None required.
`1429: Invalid PCI device group: <pci_dev_grp>` Invalid HBA PCI device group detected. Data: None Severity: Error Log: LOG_INIT Action: None required.
`1430: Failed to initialize sql list` Failed to initialize SQL list. Data: None Severity: Error Log: LOG_INIT Action: None required.
`1431: Invalid HBA PCI device group: <pci_dev_grp>` Invalid HBA PCI device group detected. Data: None Severity: Error Log: LOG_INIT Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`1432: Failed to initialize rpi headers.`

Failed to initialize RPI headers.

Data: None

Severity: Error

Log: LOG_INIT

Action: None required.

`1476: Failed to allocate sysfs attr`

Failed to allocate `sysfs` attrIbutes.

Data: None

Severity: Error

Log: LOG_INIT

Action: None required.

`1477: Failed to set up hba`

Failed to set up the HBA.

Data: None

Severity: Error

Log: LOG_INIT

Action: None required.

`1603: Loopback test did not receive expected data length. actual length <len>expected length <full_size>.`

The loopback test did not receive the same amount of data that it transmitted.

Data: None

Severity: Error

Log: LOG_LIBDFC

Action: None required.

`1800: Could not issue unreg_vpi`

Driver attempt to unregister VPI failed.

Data: None

Severity: Error

Log: LOG_VPORT verbose

Action: Software driver error. If this issue persists, report these errors to Broadcom Technical Support.

`1801: Create vport work array FAILED: cannot do scsi_host_get`

The driver was unable to get a reference to a SCSI host.

Data: None

Severity: Warning

Log: LOG_VPORT verbose

Action: Software driver warning. If this issue persists, report these errors to Broadcom Technical Support.

`1816: FLOGI NPIV supported, response data <port>`

The fabric reports support for NPIV upon FLOGI.

Data: (1) response_multiple_NPort

Severity: Warning

Log: LOG_VPORT verbose

Action: No action needed, informational.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

1817: Fabric does not support NPIV - configuring single port mode The fabric reports no support for NPIV upon FLOGI. Data: None Severity: Warning Log: LOG_VPORT verbose Action: No action needed, informational.
1818: VPort failed init, mbxCmd <mailbox command> READ_SPARM mbxStatus <mailbox status> , rc = <status> A pending mailbox command that was issued to initialize the port failed. Data: (1) mbxCommand (2) mbxStatus (3) rc Severity: Error Log: LOG_VPORT verbose Action: Software driver error. If this issue persists, report these errors to Broadcom Technical Support.
1819: Unrecognized lpfc_sli_mode parameter: <mode> An attempt was made to set the SLI mode to an invalid value. The valid values for the SLI mode are 0, 2, and 3. Data: (1) lpfc_sli_mode Severity: Error Log: LOG_VPORT verbose Action: Correct the lpfc_sli_mode driver parameter setting. Valid values are 0, 2, and 3.
1820: Unable to select SLI-3. Not supported by adapter. The HBA is incapable of operating in a given mode. Data: None Severity: Error Log: LOG_VPORT verbose Action: SLI-3 mode is only available on some HBAs. Do not attempt to force the SLI mode to 3 on HBAs that do not support SLI-3 mode. This is an informational message. HBAs that do not support SLI-3 will be configured to run in SLI-2 mode, but use the auto setting (0).
1821: Create VPORT failed. Invalid WWN format The port could not be created due to an invalid WWNN or WWPN format. Data: None Severity: Error Log: LOG_VPORT verbose Action: Provide a valid WWN when creating Vports.
1822: Invalid <name>: <xx: xx: xx: xx: xx: xx: xx: xx> An invalid WWN was used when creating a vPort. Data: (1) type_name (2) wwn[1] (3) wwn[3] (3) wwn[5] (4) wwn[7] Severity: Error Log: LOG_VPORT verbose Action: When creating a vPort, you must furnish a valid WWN.
1823: Create VPORT failed. Duplicate WWN on HBA. The port could not be created because it would duplicate an existing WWN HBA address. The resources for the port were discarded. Data: None Severity: Error Log: LOG_VPORT verbose Action: Provide a unique WWN.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`1824: NPIV enabled: Override lpfc_sli_mode parameter (<mode>) to auto(0)` The `lpfc_enable_npiv` and `lpfc_sli_mode` driver parameter settings conflict. The HBA must be configured for SLI-3 mode to support NPIV. Data: (1) lpfc_sli_mode Severity: Error Log: LOG_VPORT verbose Action: This is an informational message that indicates that the `lpfc_enable_npiv` and `lpfc_sli_mode` parameter settings are not compatible. Resolve the parameter conflict by setting the SLI mode to 0 or 3 or, if SLI-2 mode is required, disable NPIV.
`1825: Vport Created.` This message indicates that a port was created in the system. It is displayed at this level to ensure that it always appears at all log levels. Data: None Severity: Error Log: LOG_VPORT verbose Action: No action, informational.
`1826: Vport Disabled.` The port had to be disabled in the system. Data: None Severity: Error Log: LOG_VPORT verbose Action: No action, informational.
`1827: Vport Enabled` The port was enabled after possible recovery from some errors. Data: None Severity: Error Log: LOG_VPORT verbose Action: No action, informational.
`1828: Vport Deleted` A vPort was deleted. Data: None Severity: Error Log: LOG_VPORT verbose Action: No action, informational.
`1829: CT command failed to delete objects on fabric.` A command issued to the fabric to delete an associated resource for an object, such as for a port, failed. Data: None Severity: Error Log: LOG_VPORT verbose Action: Software driver error. If this issue persists, report these errors to Broadcom Technical Support.
`1830: Signal aborted mbxCmd <command>` A pending mailbox command was aborted because the thread received a signal. Data: None Severity: Error Log: LOG_VPORT verbose Action: Retry the attempted command.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`1831: Create VPORT Interrupted` The port creation process was unexpectedly interrupted at a critical time, and the operation was unsuccessful. Data: None Severity: Error Log: LOG_VPORT verbose Action: The process was interrupted while creating a vPort. Retry the command.
`1832: No pending MBOX command to handle` Data: None Severity: Error Log: LOG_MBOX Action: None required.
`1835: Vport discovery quiesce failed: state <port_state> fc_flags <fc_flag> wait msecs <jiffies_to_msecs(jiffies - start_time)>` Could not pause discovery on this vPort. Data: None Severity: Error Log: LOG_VPORT Action: None required.
`1836: Could not issue unreg_login(all_rpis) status <rc>` The unreg_login cannot be issued. Data: None Severity: Error Log: LOG_MBOX, LOG_VPORT Action: None required.
`1837: Vport_delete failed: Cannot delete static vport` Static vPorts cannot be deleted. Data: None Severity: Error Log: LOG_VPORT Action: None required.
`1838: Failed to INIT_VPI on vpi <vpi> status <rc>` Failed to INIT_VPI. Data: None Severity: Error Log: LOG_VPORT Action: None required.
`2000: Failed to allocate mbox for read_FCF cmd` Failed to allocate mailbox for READ_FCF command. Data: None Severity: Error Log: LOG_INIT Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`2001: Unable to allocate memory for issuing SLI_CONFIG_SPECIAL mailbox command` Unable to allocate memory for issuing the SLI_CONFIG_SPECIAL mailbox command. Data: None Severity: Error Log: LOG_SLI Action: None required.
`2002: Error Could not grow rpi count` An error occurred because the RPI count could not be increased. Data: None Severity: Error Log: LOG_SLI Action: None required.
`2004: Failed to allocate XRI.last XRITAG is <XRI> Max XRI is <MAX_XRI>, Used XRI is <USED_XRI>.` All XRIs are in use. Data: None Severity: Warning Log: LOG_SLI Action: None required.
`2005 - Reducing Queues - FW resource limitation: WQ %d CQ %d EQ %d: min %d: IRQ %d HDWQ %d` Reducing queues due to resource limitations. Data: None Severity: Error LOG: LOG_SLI Action: None required.
`2007: Only Limited Edition cmd Format supported <iocb.ulpCommand>` The SGL pages could not be unregistered from the firmware. Data: None Severity: Error Log: LOG_SLI Action: None required.
`2008: Error <rc> posting all rpi headers` The RPI headers could not be posted to the firmware. Data: None Severity: Error Log: LOG_SLI Action: None required.
`2009: Failed to allocate mbox for ADD_FCF cmd` Failed to allocate mailbox for `ADD_FCF` command. Data: None Severity: Error Log: LOG_INIT Action: None required.
`2010: Resume RPI Mailbox failed status <status>, mbxStatus <mbx status>` Data: None Severity: Error Log: LOG_SLI Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

2011: Unable to allocate memory for issuing SLI_CONFIG_SPECIAL mailbox command Data: None Severity: Error Log: LOG_SLI Action: None required.
2012: Mailbox failed , mbxCmd <mbx_cmd> READ_CONFIG, mbxStatus <mbx status> The READ_CONFIG mailbox command failed. Data: None Severity: Error Log: LOG_SLI Action: None required.
2013: Could not manually add FCF record 0, status <rc> Could not add an FCF record to the FCF list. Data: None Severity: Error Log: LOG_MBOX, LOG_SLI Action: None required.
2014: Invalid command <iocb.ulpCommand> The IOCB command is invalid. Data: None Severity: Error Log: LOG_SLI Action: None required.
2015: Invalid CT %x command <iocb.ulpCommand> Invalid command type in the IOCB is not supported. Data: None Severity: Error Log: LOG_SLI Action: None required.
2017: REG_FCFI mbxStatus error <mbx status> HBA state <port_state> The REG_FCFI mailbox command failed. Data: None Severity: Error Log: LOG_MBOX Action: None required.
2018: REG_VFI mbxStatus error <mbx status> HBA state <port_state> The REG_VFI mailbox command failed. Data: None Severity: Error Log: LOG_MBOX Action: None required.
2022: VPI Mailbox failed status <status>, mbxStatus <mbxStatus> The INIT-VPI mailbox command failed. Data: None Severity: Error Log: LOG_SLI Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`2500: EQ_CREATE mailbox failed with status <shdr_status> add_status <shdr_add_status>, mbx status <rc>` The mailbox command sent to create the event queue failed. Data: None Severity: Error Log: LOG_INIT Action: None required.
`2501: CQ_CREATE mailbox failed with status <shdr_status> add_status <shdr_add_status>, mbx status <rc>` The mailbox command sent to create the completion queue failed. Data: None Severity: Error Log: LOG_INIT Action: None required.
`2502: MQ_CREATE mailbox failed with status <shdr_status> add_status <shdr_add_status>, mbx status <rc>` The mailbox command sent to create the mailbox queue failed. Data: None Severity: Error Log: LOG_INIT Action: None required.
`2503: WWQ_CREATE mailbox failed with status <shdr_status> add_status <shdr_add_status>, mbx status <rc>` The mailbox command sent to create the work queue failed. Data: None Severity: Error Log: LOG_INIT Action: None required.
`2504: RQ_CREATE mailbox failed with status <shdr_status> add_status <shdr_add_status>, mbx status <rc>` The mailbox command sent to create the RQ failed. Data: None Severity: Error Log: LOG_INIT Action: None required.
`2505: EQ_DESTROY mailbox failed with status <shdr_status> add_status <shdr_add_status>, mbx status <rc>` The mailbox command sent to delete the event queue failed. Data: None Severity: Error Log: LOG_INIT Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`2506: CQ_DESTROY mailbox failed with status <shdr_status> add_status <shdr_add_status>, mbx status <rc>` The mailbox command sent to delete the completion queue failed. Data: None Severity: Error Log: LOG_INIT Action: None required.
`2507: MQ_DESTROY mailbox failed with status <shdr_status> add_status <shdr_add_status>, mbx status <rc>` The mailbox command sent to delete the mailbox queue failed. Data: None Severity: Error Log: LOG_INIT Action: None required.
`2508: WQ_DESTROY mailbox failed with status <shdr_status> add_status <shdr_add_status>, mbx status <rc>` The mailbox command sent to delete the work queue failed. Data: None Severity: Error Log: LOG_INIT Action: None required.
`2509: RQ_DESTROY mailbox failed with status <shdr_status> add_status <shdr_add_status>, mbx status <rc>` The mailbox command sent to delete the work queue failed. Data: None Severity: Error Log: LOG_INIT Action: None required.
`2510: RQ_DESTROY mailbox failed with status <shdr_status> add_status <shdr_add_status>, mbx status <rc>` The mailbox command sent to delete the work queue failed. Data: None Severity: Error Log: LOG_INIT Action: None required.
`2511: POST_SGL mailbox failed with status <shdr_status> add_status <shdr_add_status>, mbx status <rc>n` The mailbox command sent to post the SGL pages to the firmware failed. Data: None Severity: Error Log: LOG_INIT Action: None required.
`2512 - MODIFY_EQ_DELAY mailbox failed with status x%x add_status x%x, mbx status x%x` The mailbox command sent to modify EQ delay failed. Data: None Severity: Error LOG: LOG_INIT Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

2513: POST_SGL_BLOCK mailbox command failed status <shdr_status> add_status <shdr_add_status> mbx status <rc> The mailbox command sent to post the SGL pages to the firmware failed. Data: None Severity: Error Log: LOG_INIT Action: None required.
2514: POST_RPI_HDR mailbox failed with status <shdr_status> add_status <shdr_add_status>, mbx status <rc> The mailbox command sent to post the RPI header pages to the firmware failed. Data: None Severity: Error Log: LOG_INIT Action: None required.
2515: ADD_FCF_RECORD mailbox failed with status <rc> The mailbox command to add the FCF record failed. Data: None Severity: Error Log: LOG_INIT Action: None required.
2516 - CGN SET_FEATURE mbox failed with status x%x add_status x%x, mbx status x%x Reset congestion to FPINs only. Data: None Severity: Error LOG: LOG_INIT | LOG_CGN_MGMT Action: None required.
2517 - CGN Signal Event - Warn: x%x Alarm: x%x Data: None Severity: Error LOG: LOG_CGN_MGMT Action: None required.
2518 - EEPROM failure - Event Data1: x%08x Event Data2: EEPROM failure occurred. Data: event_data1, event_data2 Severity: Error LOG: LOG_SLI Action: Contact Broadcom Technical Support.
2521: READ_FCF_RECORD mailbox failed with status <shdr_status> add_status <shdr_add_status>, mbx The READ_FCF_RECORD mailbox command failed. Data: None Severity: Error Log: LOG_INIT Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`2523: Allocated DMA memory size (<alloc_len>) is less than the requested DMA memory size (<req_len>)` The `ADD_FCF_RECORD` mailbox command failed to retrieve the length required from the firmware. Data: None Severity: Error Log: LOG_INIT Action: None required.
`2524: Failed to get the non-embedded SGE virtual address` The `READ_FCF_RECORD` mailbox command could not retrieve the SGE that was requested. Data: None Severity: Error Log: LOG_MBOX Action: None required.
`2527: Failed to allocate non-embedded SGE array.` Failed to allocate the nonembedded SGE array. Data: None Severity: Error Log: LOG_MBOX Action: None required.
`2528: Mailbox command <vpi> cannot issue` The mailbox command could not be issued because the mailbox interrupt is disabled. Data: (1) mbxCommand (2) sli_flag (3) flag Severity: Error Log: LOG_MBOX, LOG_SLI Action: None required.
`2529: Mailbox command <vpi> cannot issue` Data: (1) mbxCommand (2) sli_flag (3) flag Severity: Error Log: LOG_MBOX, LOG_SLI Action: None required.
`2530: Mailbox command <vpi> cannot issue` The SLI layer in the driver is inactive. Data: (1) mb.mbxCommand (2) sli_flag (3) flag Severity: Error Log: LOG_MBOX, LOG_SLI Action: None required.
`2531: Mailbox command <cpi> cannot issue` Data: (1) mb.mbxCommand (2) sli_flag (3) flag Severity: Error Log: LOG_MBOX, LOG_SLI Action: None required.
`2532: Mailbox command <vpi> (<mbxCommand>) cannot issue` The mailbox bootstrap code detected that the SLI layer is active. Data: (1) sli4_mbox_opcode (2) sli_flag,(3) MBX_POLL Severity: Error Log: LOG_MBOX, LOG_SLI Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

2533: Mailbox command <vpi> (<mbxCommand>) cannot issue Data: (1) sli4_mbox_opcode (2) sli_flag (3) MBX_NOWAIT Severity: Error Log: LOG_MBOX, LOG_SLI Action: None required.
2535: Unsupported RQ count. (<entry_count>) The RQ ring can only be 512, 1024, 2048, or 4096. Data: None Severity: Error Log: LOG_SLI Action: None required.
2536: Unsupported RQ count. (<entry_count>) The RQ ring can only be 512, 1024, 2048, or 4096. Data: None Severity: Error Log: LOG_SLI Action: None required.
2537: Receive Frame Truncated! The receive unsolicited handler detected a truncated frame. Data: None Severity: Error Log: LOG_SLI Action: None required.
2539: Dropped frame rctl:%s type:%s\n An unsupported frame was received by the port and dropped. Data: (1) rctl_names[fc_hdr->fh_r_ctl], (2) type_names[fc_hdr->fh_type] Severity: Error Log: Always Action: No action needed, informational.
2540: Ring <ring #> handler: unexpected Rctl <fh_rctl> Type <fh_type> The received frame has an unsupported RCTL or FH_TYPE. Data: None Severity: Warning Log: LOG_SLI Action: None required.
2541: Mailbox command <vpi> (<mbxCommand>) cannot issue Data: (1) sli_mbox_opcode (2) sli_flag (3) flag Severity: Error Log: LOG_MBOX, LOG_SLI Action: None required.
2542: Try to issue mailbox command <vpi> (<mbxCommand>) synchronously ahead of async mailbox command queue Attempting to send a synchronous mailbox command ahead of the asynchronous mailbox commands. Data: (1) sli4_mbx_opcode or sli_mbox_opcode, (2) sli_flag, (3) flag Severity: Warning Log: LOG_MBOX, LOG_SLI Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`2543: Mailbox command <vpi> (<mbxCommand>) cannot issue` The mailbox command does not have all of the fields set correctly. Data: (1) sli_mbox_opcode (2) sli_flag (3) flag Severity: Error Log: LOG_MBOX, LOG_SLI Action: None required.
`2544: Mailbox command <vpi> (<mbxCommand>) cannot issue` The HBA cannot be accessed on the PCI bus. Data: (1) sli_mbox_opcode (2) sli_flag (3) flag Severity: Error Log: LOG_MBOX, LOG_SLI Action: None required.
`2546: New FCF found index <index> tag <event_tag>` A new FCF is found. Data: None Severity: Error Log: LOG_DISCOVERY Action: None required.
`2547: Read FCF record failed` Could not read the FCF record from the firmware. Data: None Severity: Error Log: LOG_DISCOVERY Action: None required.
`2548: FCF Table full count <count> tag <event_tag>` The FCF table is full. Data: None Severity: Error Log: LOG_SLI Action: None required.
`2549: FCF disconnected from network index <index> tag <event_tag>` The FCF disconnected from the network. Data: None Severity: Error Log: LOG_DISCOVERY Action: None required.
`2550: UNREG_FCFI mbxStatus error <u.mb.mbxStatus> HBA state <port_state>.` The `UNREG_FCFI` mailbox command failed. Data: None Severity: Error Log: LOG_DISCOVERY. LOG_MBOX Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`2551: UNREG_FCFI mbox allocation failed HBA state <port_state>` The allocation for the `UNREG_FCFI` mailbox command failed. Data: None Severity: Error Log: LOG_DISCOVERY, LOG_MBOX Action: None required.
`2552: UNREG_FCFI issue mbox failed rc <rc> HBA state <port_state>` The `UNREG_FCFI` mailbox command failed. Data: None Severity: Error Log: LOG_DISCOVERY, LOG_MBOX Action: None required.
`2553: lpfc_unregister_unused_fcf failed to read FCF record HBA state` Data: None Severity: Error Log: LOG_DISCOVERY, LOG_MBOX Action: None required.
`2554: Could not allocate memory for fcf record` Data: None Severity: Error Log: LOG_MBOX, LOG_SLI Action: None required.
`2555: UNREG_VFI mbxStatus error <u.mb.mbxStatus> HBA state <port_state>` The `UNREG_VFI` mailbox command failed. Data: None Severity: Error Log: LOG_DISCOVERY, LOG_MBOX Action: None required.
`2556: UNREG_VFI mbox allocation failed HBA state <port_state>` Could not allocate memory for `UNREG_VFI` mailbox command. Data: None Severity: Error Log: LOG_DISCOVERY, LOG_MBOX Action: None required.
`2557: UNREG_VFI issue mbox failed rc <rc> HBA state <port_state>` Could not issue the `UNREG_VFI` mailbox command. Data: None Severity: Error Log: LOG_DISCOVERY, LOG_MBOX Action: None required.
`2558: ADD_FCF_RECORD mailbox failed with status<shdr_status> add_status <shdr_add_status>` The `ADD_FCF_RECORD` mailbox command failed. Data: None Severity: Error Log: LOG_INIT Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

2559: Block sgl registration required DMA size <reqlen> great than a page. Attempting to register more SGEs with the firmware than can fit in a page. Data: None Severity: Warning Log: LOG_INIT Action: None required.
2562 - Failure to allocate an ELS sgl entry:%d Memory allocation failed for SGL entry. Data: None Severity: Error LOG: LOG_SLI Action: None required.
2563 - Failure to allocate an ELS mbuf:%d Memory allocation failed for ELS buffer. Data: None Severity: Error LOG: LOG_SLI Action: None required.
2566: Failed to allocate table entry Failed to allocate connection table entry. Data: None Severity: Error Log: LOG_INIT Action: None required.
2567: Config region 23 has bad signature The driver was unable to read Config Region 23 because it has an invalid signature. Data: None Severity: Error Log: LOG_INIT Action: None required.
2568: Config region 23 has bad version The driver was unable to read Config Region 23 because it is an invalid version. Data: None Severity: Error Log: LOG_INIT Action: None required.
2572: Failed allocate memory for fast-path per-EQ handle array Failed to allocate memory for the fast-path per-EQ handle array. Data: None Severity: Error Log: LOG_INIT Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

2574 - IO channels: hdwQ %d IRQ %d MRQ: %d Information about HWQ, interrupts, and MRQ. Data: None Severity: Error LOG: LOG_INIT Action: None required.
2579: Slow-path wqe consume event carries miss-matched qid: wcqe-qid=<wcqe_quid>, sp-qid=<sp_quid> The consumed entry does not have the queue ID of the slow-path. Data: None Severity: Warning Log: LOG_SLI Action: None required.
2580: Fast-path wqe consume event carries miss-matched qid: wcqe-qid=<fcp_wqid>. The consumed entry does not have the queue ID of the fast path. Data: None Severity: Warning Log: LOG_SLI Action: None required.
2597 - Sync Mailbox command x%x (x%x/x%x) failure: mqe_sta: x%x mcqe_sta: x%x/x%x Data: x%x x%x Issue of synchronous mailbox command failed. Data: None Severity: Error LOG: LOG_MBOX | LOG_SLI Action: None required.
2598: Adapter Link is disabled. The adapter link is disabled. Data: None Severity: Error Log: LOG_INIT Action: None required.
2599: Adapter failed to issue DOWN_LINK mbox command rc <rc> The driver was unable to issue the DOWN_LINK mailbox command. Data: None Severity: Error Log: LOG_INIT Action: None required.
2600 - failed to allocate mailbox memory Memory allocation failed. Severity: Error Data: None LOG: LOG_INIT Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`2605: lpfc_dump_static_vport: memory allocation failed` Failed to allocate mailbox memory. Data: None Severity: Error Log: LOG_MBOX Action: None required.
`2606: No NPIV Fabric support` No NPIV Fabric support. Data: None Severity: Error Log: LOG_ELS Action: None required.
`2607: Failed to allocate init_vpi mailbox` Failed to allocate the `INIT_VPI` mailbox command. Data: None Severity: Error Log: LOG_MBOX Action: None required.
`2608: Failed to issue Init VPI mailbox` The driver was unable to send an `INIT_VPI` mailbox command. Data: None Severity: Error Log: LOG_MBOX Action: None required.
`2609: Init VPI mailbox failed <u.mb.mbxStatus>` The `INIT_VPI` mailbox command failed. Data: None Severity: Error Log: LOG_MBOX Action: None required.
`2610 - FCF (x%x) reached driver's book keeping dimension:x%x` FCF index exceeded maximum FCF table index. Data: None Severity: Error LOG: LOG_FIP Action: None required.
`2611 - FLOGI failed on FCF (x%x), status:x%x/x%x, tmo:x%x, perform roundrobin FCF failover` FLOGI failed on FCF. Data: None Severity: Error LOG: LOG_FIP | LOG_ELS Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`2619: Config region 23 has bad signature` Configuration region 23 has an invalid signature. Data: None Severity: Error Log: LOG_INIT Action: None required.
`2620: Config region 23 has bad version` Configuration region 23 has an invalid version. Data: None Severity: Error Log: LOG_INIT Action: None required.
`2621 - IO completion: Not an active IO` The I/O was not an active I/O in the completion routine. Data: None Severity: Error LOG: LOG_INIT Action: None required.
`2707: Ring <Ring#> handler: Failed to allocate iocb Rctl <fh_rctl> Type <fh_type> received` The driver was unable to allocate memory to send a query configuration mailbox command. Data: None Severity: Error Log: LOG_SLI Action: None required.
`2717: CT context array entry [<index>] over-run: oxid:<fh_ox_id>, sid:<fh_SID>` All of the array slots to hold buffers that are passed to the application are in use. Data: None Severity: Warning Log: LOG_ELS Action: None required.
`2718: Clear Virtual Link Received for VPI <index> tag <event_tag>` A clear virtual link was received from the fabric for this VPI. Data: None Severity: Error Log: LOG_DISCOVERY Action: None required.
`2719: Invalid response length: tgt <TGT_ID> lun <LUN> cmnd <CMD> rsplen <RSPLEN>` The response length for this FCP command is not supported. Data: None Severity: Error Log: LOG_FCP Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

`2721: ndlp null for oxid %x SID %x\n, icmd->ulpContext, dfchba->ct_ctx[tag].SID);` The node value for this SID is not in the node list. Data: None Severity: Warning Log: LOG_ELS Action: None required.
`2726: READ_FCF_RECORD Indicates empty FCF table` The driver requested the firmware for a list of FCF entries to connect to and the firmware responded indicating that the FCF table is empty. Data: None Severity: Error Log: LOG_INIT Action: None required.
`6045: Issue GEN REQ WQE to NPORT <nlp_DID> Data <iotag> <port_state>` Not able to queue NVMe LS command. Data: (1) nlp_DID, (2) iotag and (3) port_state. Severity: Error Log: LOG_ELS Action: Reload the driver.
`6088: DID x<port_id> not ready for IO. State <nlp_state>, Type <nlp_type>` Target port it not ready. Data: (1) port_id, (2) nlp_state and (3) nlp_type. Severity: Error Log: LOG_NODE Action: Verify that all targets are still visible.
`6051: Remoteport x<pnvme_rport>, rport has invalid ndlp. Failing LS Req` Target port it not ready. Data: pnvme_rport Severity: Error Log: LOG_NODE Action: Verify that all targets are still visible.
`6044: Could not find node for DID <port_id>` Not able to allocate memory for NVMe LS request. Data: port_id Log: LOG_NVME_DISC Severity: Error Action: This message indicates one of these two issues: (1) Kernel virtual memory is depleted. Check that the system meets minimum RAM requirements for the adapter. Try closing other applications to free some memory. (2) A possible driver buffer management issue. If this issue persists, report the error to Broadcom Technical Support.
`6081: NVME Completion Protocol Error: xri <sli4_xritag> status x<status> result x<result> placed x<total_data_placed>` NVMe I/O completed with a protocol error. Data: (1) sli4_xritag, (2) status, (3) result and (4) total_data_placed. Log: LOG_NVME_IOERR Severity: Error Action: If this issue persists, check the targets. If the targets are not the problem, report the error to Broadcom Technical Support.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

6072: NVME Completion Error: xri <sli4_xritag> status x<status> result x<result> [x<parameter>] placed x<total_data_placed> NVMe I/O completed with an error. Data: (1) sli4_xritag, (2) status, (3)result ,(4) parameter and (4) total_data_placed. Log: LOG_NVME_IOERR Severity: Info Action: This error could indicate a software driver, firmware, or hardware issue. If this issue persists, report the error to Broadcom Technical Support.
6058: Too many sg segments from NVME Transport. Max <cfg_nvme_seg_cnt>, nvmeIO sg_cnt <seg_cnt> Too many sg segments for an NVMe command request. Data: (1) cfg_nvme_seg_cnt and (2) seg_cnt. Log: LOG_NVME_IOERR Severity: Error Action: This error could indicate a software driver issue. If this issue persists, report the error to Broadcom Technical Support.
6052: EXIT. issue ls wqe failed lport x<pnvme_lport>, rport x<pnvme_rport> lsreq x<pnvme_lsreq> Status <ret> DID <nlp_DID> Not able to queue NVMe LS command. Data: (1) pnvme_lport>, (2) pnvme_rport, (3) pnvme_lsreq, (4) ret and (5) nlp_DID. Log: LOG_NVME_DISC Severity: Error Action: This error could indicate a software driver, firmware, or hardware issue. If this issue persists, report the error to Broadcom Technical Support.
6049: Could not find node for DID <port_id> Not able to abort LS request. Data: port_id Log: LOG_NVME_ABTS Severity: Error Action: This error could indicate a software driver issue. If this issue persists, report the error to Broadcom Technical Support.
6071: Null lpfc_ncmd pointer. No release, skip completion Skip NVMe I/O completion because of null lpfc_ncmd pointer. Data: None Log: LOG_NVME_IOERR Severity: Error Action: This error could indicate a software driver issue. If this issue persists, report the error to Broadcom Technical Support.
6066: Missing cmpl ptrs: lpfc_ncmd x<lpfc_ncmd>, nvmeCmd x<nvmeCmd> Missing completion pointer. Data: (1) lpfc_ncmd and (2) nvmeCmd. Log: LOG_NVME_IOERR Severity: Error Action: This error could indicate a software driver issue. If this issue persists, report the error to Broadcom Technical Support.
6062 Ignoring NVME cmpl. No ndlp Ignoring NVMe completion, no NDLP. Data: None Log: LOG_NVME_IOERR Severity: Error Action: This error could indicate a software driver issue. If this issue persists, report the error to Broadcom Technical Support.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

6059: dptr err <i>, nseg <nseg> Scatter-gather data pointer is null. Data: (1) i and (2) nseg. Log: LOG_NVME_IOERR Severity: Error Action: This error could indicate a software driver issue. If this issue persists, report the error to Broadcom Technical Support.
6063: NVME DMA Prep Err: sg_cnt <sg_cnt> payload_length x<payload_length> NVMe DMA preparation failed. Data: (1) sg_cnt and (2) payload_lengthLog: LOG_NVME_IOERR Severity: Error Action: This error could indicate a software driver issue. If this issue persists, report the error to Broadcom Technical Support.
6139: Driver in reset cleanup - flushing NVME Req now. hba_flag <hba_flag> Driver in reset cleanup. Data: hba_flag Log: LOG_NVME_ABTS Severity: Error Action: None required.
6140: NVME IO req <freqpriv> has no matching lpfc nvme io buffer. Skipping abort req. No matching NVMe I/O buffer to abort, skipping abort request. Data: freqpriv Log: LOG_NVME_ABTS Severity: Error Action: This error could indicate a software driver issue. If this issue persists, report the error to Broadcom Technical Support.
6141: lpfc NVME IO req has no nvme_fcreq io buffer. Skipping abort req No nvmeCmd for NVMe I/O buffer to abort, skipping abort request. Log: LOG_NVME_ABTS Severity: Error Action: None required.
6143: NVME req mismatch: lpfc_nbuf x<lpfc_nbuf> nvmeCmd x<nvmeCmd>, pnvme_fcreq x<pnvme_fcreq>. Skipping Abort xri x<sli4_xritag> NVMe I/O abort request mismatch between FC NVMe command and transport pnvme_fcreq. Data: (1) lpfc_nbuf, (2) nvmeCmd, (3) pnvme_fcreq and (4) sli4_xritag. Log: LOG_NVME_ABTS Severity: Error Action: This error could indicate a software driver issue. If this issue persists, report the error to Broadcom Technical Support.
6142: NVME IO req x<pnvme_fcreq> not queued - skipping abort req xri x<sli4_xritag> NVMe I/O request pnvme_fcreq not queued, skipping abort request. Data: (1) pnvme_fcreq and (2) sli4_xritag. Log: LOG_NVME_ABTS Severity: Error Action: None required.

Table 8: FC Error Log Messages and Their Descriptions (Continued)

6144: Outstanding NVME I/O Abort Request still pending on nvme_fcreq x<pnvme_fcreq>, lpfc_ncmd <lpfc_nbuf> xri x<sli4_xritag> Outstanding NVMEe I/O abort request still pending. Data: (1) pnvme_fcreq and (2) sli4_xritag. Log: LOG_NVME_ABTS Severity: Error Action: None required.
6176: Lport x<lport> Localport x<localport> wait timed out. Pending <pending>. Renewing. NVMe port is waiting for I/O completion. Data: (1) lport, (2) localport and (3) pending. Log: LOG_NVME_IOERR Severity: Error Action: This error could indicate a software driver issue. If this issue persists, report the error to Broadcom Technical Support.
6031: RemotePort Registration failed err: <ret>, DID x<nlp_DID> RemotePort registration with NVMe transport failed. Data: (1) ret and (2) nlp_DID. Log: LOG_NVME_DISC Severity: Error Action: This error could indicate a software driver issue. If this issue persists, report the error to Broadcom Technical Support.
6172: NVME rescanned DID x<nlp_DID> port_state x<port_state> NVMe rescanned for an NVMe target. Data: (1) nlp_DID and (2) port_state. Log: LOG_NVME_DISC Severity: Error Action: None required.
6167: NVME unregister failed <ret> port_state x<port_state> Unregister NVMe target port with NVMe transport failed. Data: (1) ret and (2) port_state. Log: LOG_NVME_DISC Severity: Error Action: None required.
6168: State error: lport x<localport>, rport x<rport> FCID <nlp_DID> Uregister NVMe target port failed. Data: (1) localport, (2) rport and (3) nlp_DID. Log: LOG_NVME_DISC Severity: Error Action: None required.

Appendix A: lpfc Driver BlockGuard Functionality

This appendix describes how to enable BlockGuard and set `lpfc` driver module parameters.

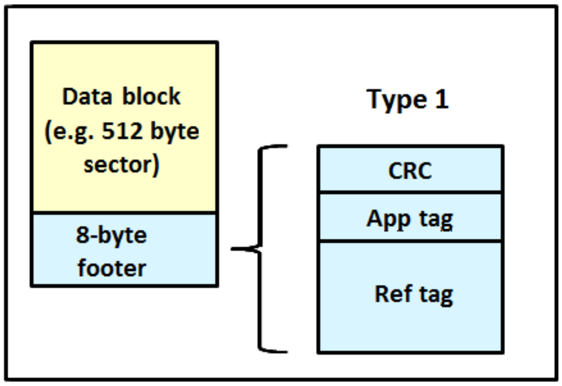
A.1 Overview

The BlockGuard feature checks the integrity of data read from and written to the host to the disk and back through the SAN. This check is implemented through the DIF defined in the ANSI T10 standard.

The Emulex `lpfc` driver supports T10 DIF Type 1. In the Type 1 implementation, the 8-byte DIF consists of a Ref tag (or LBA), an App tag, and a Guard tag (or CRC). A Type 1 DIF is defined as a having a 2-byte Guard tag, a 2-byte App tag, and a 4-byte Ref tag, which consists of the lower 32 bits of the logical block address.

The following figure shows a data block (with a 512-byte sector) with the 8-byte footer attached to the end. The contents of the 8-byte footer are shown with the fields that make up the Type 1 DIF: the Guard tag, the App tag, and the Ref tag. The App tag is not used by the `lpfc` driver.

Figure 1: Data Block Showing the Type 1 DIF



When data is written, the DIF is generated by the host, or by the adapter, based on the block data and the LBA. The DIF field is added to the end of each data block, and the data is sent through the SAN to the storage target. The storage target validates the CRC and Ref tag and, if correct, stores the data block and DIF on the physical media. If the CRC does not match the data, the data was corrupted during the write. A Check Condition is returned to the host with the appropriate error code. The host records the error and retransmits the data to the target. In this way, data corruption is detected immediately on a write and never committed to the physical media. On a read, the DIF, along with the data block, is returned to the host, which validates the CRC and Ref tags. Because this validation is done by the hardware, it adds a very small amount of latency to the I/O.

The format of the Guard tag can optionally be an IP checksum instead of the CRC mandated by the T10 DIF. This setting can be beneficial because the initiator host uses less CPU overhead to generate an IP checksum than it does with a CRC. The IP checksum is typically passed as the Guard tag between the initiator host and the adapter. The adapter hardware translates the IP checksum into a CRC, or visa versa, on data being sent and received on the wire. The CRC is called a DIF protection type, and the IP checksum is referred to as a DIX protection type.

A.2 Enabling BlockGuard

BlockGuard is disabled by default. To enable it, the `lpfc_enable_bg` parameter must be passed to the driver as follows:

```
insmod lpfc.ko lpfc_enable_bg=1
```

For a permanent configuration that will persist across system reboots, create the `/etc/modprobe.d/elx-lpfc.conf` file, and place the following line into it:

```
options lpfc lpfc_enable_bg=1
```

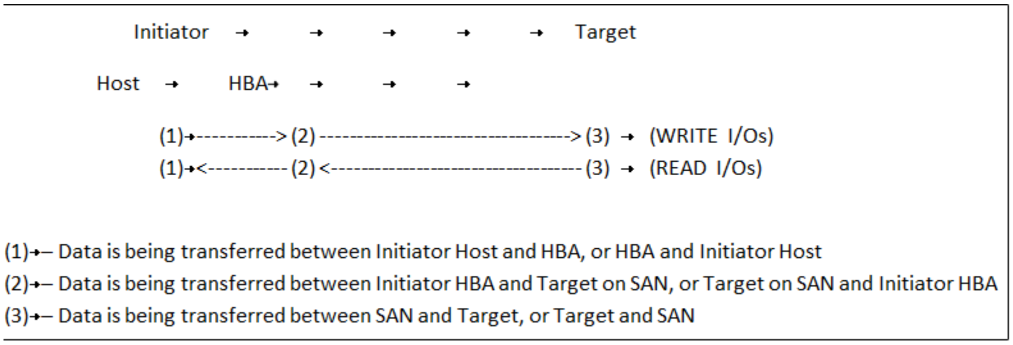
Additional module parameters can be added to this line, separated by spaces.

A.3 SCSI Command Support – SCSI Operation Codes

When there are both `READ` and `WRITE` requests, the CDB passed to the adapter from the initiator host has a read protect/ write protect (`RDPROTECT`/`WRPROTECT`) field that indicates to the target whether to perform data integrity verification. It also indicates whether to transfer protection data between initiator and target. The adapter does not know if a target supports protection information or with which type of protection it is formatted. The initiator host, which has this knowledge, will always prepare a CDB with the appropriate `RDPROTECT`/`WRPROTECT` information, depending on target format and capabilities. The request will also include information about the protection type with which the target has been formatted.

In addition, the initiator host will also provide the adapter with an operation code that tells the controller how to place the protection data for the type of I/O to perform. Each I/O is logically a two-step process. The data is transferred between the initiator host and the adapter (over the PCI bus) and between the adapter and the target (over the SAN) as shown in the following figure. The type of operation defines whether the data transfer has protection data.

Figure 2: Data Transfer between the Initiator Host and the Adapter


Initiator → → → → → Target
Host → HBA→ → → →
(1)→----------->(2)---------------------------------->(3) → (WRITE I/Os)
(1)→<-----------(2)<----------------------------------(3) → (READ I/Os)

(1)→– Data is being transferred between Initiator Host and HBA, or HBA and Initiator Host
(2)→– Data is being transferred between Initiator HBA and Target on SAN, or Target on SAN and Initiator HBA
(3)→– Data is being transferred between SAN and Target, or Target and SAN

The initiator operations are listed in the following table.

Table 9: Initiator Operations

Initiator Operation	Initiator Host <-> Adapter	Adapter <-> Target	Comment
NORMAL	Unprotected	Unprotected	Used for unprotected I/O.
READ_INSERT	Protected	Unprotected	Reads the data from the target. The adapter then generates the protection data and transfers both data and protection data to the initiator host. No protection data is sent on the SAN. The adapter can insert the protection data Guard tag as CRC or IP CSUM.
READ_PASS	Protected	Protected	Reads the data and protection data from the target on the SAN. The adapter will verify data integrity and transfer both data and protection data to the initiator host. The adapter can convert the protection data Guard tag from CRC to IP CSUM.
READ_STRIP	Unprotected	Protected	Reads data and protection data from the target. The adapter will verify data integrity, discard protection data, and transfers only the data to the initiator host. It does not send the protection data to the initiator host. Protection data is sent only on the SAN.
WRITE_INSERT	Unprotected	Protected	Transfers the data from the initiator host. The adapter will then generate protection data and write both the data and protection data to the target. Protection data is sent only on the SAN.
WRITE_PASS	Protected	Protected	Transfers the data and protection data from the initiator host to the adapter. The adapter will verify protection data and write both data and protection data to the target on the SAN. The adapter can convert the protection data Guard tag from IP CSUM to CRC.
WRITE_STRIP	Protected	Unprotected	Transfers data and protection data from the initiator host. The adapter will verify data integrity, discard protection data, and writes the data only to the target. No protection data is sent on the SAN.

A.4 lpfc Driver Module Parameters

The `lpfc` driver has two module parameters: `lpfc_prot_mask` and `lpfc_prot_guard`. Using these parameters, you can control which DIF capabilities the `lpfc` driver registers with the Linux SCSI subsystem. This, in turn, controls which initiator operations (BlockGuard profiles) are used during I/O operations. These parameters are set up when the driver loads and cannot be changed while the driver is running.

A.4.1 lpfc_prot_mask

The lpfc_prot_mask parameter controls the DIF operations that the driver registers with the operating system. The operating system selects an operation to use for each I/O command that matches the adapter DIF capability. The driver indicates its capabilities by the operations it registers with the operating system.

If the parameter is not passed to the driver, the default DIX_TYPE0 is used.

The SCSI layer will typically use the bit masks listed in the following table to determine how to place the protection data associated with I/Os to the SCSI host. The default value for lpfc_prot_mask is to allow all of the options.

Table 10: lpfc_prot_mask Protection Types

Flag	Value	Indicates	Description
SHOST_DIF_TYPE1_PROTECTION	1	Adapter supports T10 DIF Type 1	Adapter-to-target Type 1 protection
SHOST_DIX_TYPE0_PROTECTION	8	Adapter supports DIX Type 0	Host-to-adapter protection only
SHOST_DIX_TYPE1_PROTECTION	16	Adapter supports DIX Type 1	Host-to-adapter Type 1 protection

The following table shows how protection data gets placed for each supported profile.

Table 11: Protection Data Placement for Supported Profiles

Flag	Value	BlockGuard Profile	Operation
SHOST_DIF_TYPE1_PROTECTION	1	A1	READ_STRIP / WRITE_INSERT
SHOST_DIX_TYPE0_PROTECTION	8	AST2	READ_INSERT / WRITE_STRIP
SHOST_DIX_TYPE1_PROTECTION SHOST_DIF_TYPE1_PROTECTION	17	AST1 / C1	READ_PASS / WRITE_PASS

A.4.2 lpfc_prot_guard

The lpfc_prot_guard parameter correlates to the SCSI_host_guard_type of the Linux kernel. This parameter specifies the type of CRC that the Linux operating system will pass to the lpfc driver. The following table shows the two guard types: CRC and IP CSUM, with values of 0x1 and 0x2, respectively.

Table 12: lpfc_prot_guard Guard Types

Flag	Value	Indicates
SHOST_DIX_GUARD_CRC	0x1	Adapter supports T10 DIF CRC
SHOST_DIX_GUARD_IP	0x2	Adapter supports both T10 DIF CRC and IP CSUM

The default value for lpfc_prot_guard is SHOST_DIX_GUARD_IP. This value defines the format for the Guard tag when the data is transferred between the host and the adapter. When the data is transferred on the wire, the protection data Guard tag is always translated into a T10 DIF CRC. To override the default, you can pass a module parameter value with either insmod or modprobe.

The SCSI layer typically uses an IP CSUM as the method for computing the protection data guard tag because it uses less CPU overhead.

Appendix B: Configuring NVMe over FC on a Target

NOTE:

- Before configuring NVMe over FC on a target using native NVMe CLI commands, ensure that you have installed the latest Emulex firmware for the LPe31000-series, LPe32000-series, and LPe35000-series adapters.
- At this time, NVMe over FC on target devices is supported on SLES 12 SP4 only.
- The NVMe target drivers are meant for evaluation and testing during development only. They are not intended for production use.

B.1 Configuring the elx-lpfc.conf File

To configure the `elx-lpfc.conf` file, perform the following steps:

1. Find the WWPNs of the HBAs that are available to be used as NVMe target ports by typing the following command:

   ```
   cat /sys/class/fc_host/host*/<port_name>
   ```

   Information similar to the following is displayed (the low number is usually port 0 of the adapter):

   ```
   0x10000090fa931110
   0x10000090fa931111
   ```

   Record the WWPNs of the adapters you want to use as target ports; you will need this information in Step 5.

2. Find the WWNNs of the HBAs in Step 1 by typing the following command:

   ```
   cat /sys/class/fc_host/host*/<node_name>
   ```

   Information similar to the following is displayed:

   ```
   0x20000090fa931110
   0x20000090fa931111
   ```

   Record the WWNNs of the adapters that you want to use as target ports; you will need this information in Step 9.

3. NVMe subsystems and namespaces are created from the storage block devices on the target. An existing storage block device on the target can be used by NVMe namespaces as their storage media. Display the available storage block devices to share over the NVMe fabric by typing the following command:

   ```
   lsblk
   ```

   All block devices found by the target are displayed, for example:

```
 NAME                                         MAJ:MIN  RM   SIZE     RO   TYPE    MOUNTPOINT
sda                                             8:0    0    136.7G   0    disk
├─sda1                                          8:1    0    2G       0    part
├─sda2                                          8:2    0    40G      0    part
├─sda3                                          8:3    0    94.7G    0    part
└─3600508b1001037383941424344450400           254:0    0    136.7G   0    mpath
  ├─3600508b1001037383941424344450400-part1   254:1    0    2G       0    part    [SWAP]
  ├─3600508b1001037383941424344450400-part2   254:2    0    40G      0    part    /var/opt
  └─3600508b1001037383941424344450400-part3   254:3    0    94.7G    0    part    /home
sdb                                             8:16   0    3G       0    disk
  └─3600000e00d2a0000002a017300340000         254:4    0    3G       0    mpath
sdc                                             8:32   0    3G       0    disk
```

```
  └─3600000e00d2a0000002a017300350000          254:5   0    3G        0    mpath
sdd                                             8:48   0    3G        0    disk
  └─3600000e00d2a0000002a017300360000          254:6   0    3G        0    mpath
sde                                             8:64   0    3G        0    disk
  └─3600000e00d2a0000002a017300370000          254:7   0    3G        0    mpath
sdf                                             8:80   0    3G        0    disk
  └─3600000e00d2a0000002a017300380000          254:8   0    3G        0    mpath
```

Record the block devices that you want to use, for example `/dev/sde`, `/dev/sdf`, or `/dev/nvme0n1`.

NOTE: Block devices are located in the `/dev` directory.

4. Navigate to the `/etc/modprobe.d` subdirectory, and create a file with the driver name `elx-lpfc.conf`.
5. Insert the following line in the `elx-lpfc.conf` file:

   ```
   options lpfc lpfc_enable_nvmet=<WWPN1, WWPN2> lpfc_enable_fc4_type=3
   ```

   where <*WWPN1, WWPN2*> are the WWPNs from Step 1.

NOTE: By default the SCSI and NVMe over FC protocols are enabled on the port and the port is an initiator for the SCSI protocol. The WWPNs that are listed indicate FC ports that are exclusively NVMe over FC targets; the SCSI protocol is not used on these ports.

6. Save the `elx-lpfc.conf` file.
7. Regenerate the ramdisk for the server by typing the following command:

   ```
   #dracut --force
   ```

8. Reboot the system.

   After the system is rebooted, the configured protocols are enacted on the FC ports, and the ports identified by the WWPNs in Step 5 are now running as target ports.

B.2 Configuring Subsystems, Namespaces, and Ports

This procedure creates target subsystems, namespaces, and ports and is for example purposes only. Refer to any operating system or community-provided NVMe CLI documentation on how to use `nvmetcli` to create NVMe target subsystems with all the potential features. For example, although newer NVMe CLI implementations support ANA attributes, the procedure that follows does not attempt to manage ANA group defaults or assignments for the subsystems or ports. If ANA is desired, refer to the operating system or community documentation for the necessary steps.

Use the NVMe CLI for target systems to create the target subsystems, namespaces, and ports, and save them to a `.json` file by performing the following procedure:

1. Start the NVMe CLI for target systems by typing the following command:

   ```
   nvmetcli
   ```

NOTE: A warning message might appear the first time you run this command. Ignore the message.

A separate utility screen appears. Step 2 though Step 13 are performed in this screen.

2. Navigate to the `/subsystems` directory, and create a subsystem, such as `fctest1`, by typing the following command:

   ```
   create fctest1
   ```

3. Navigate to the `/subsystems/fctest1` subdirectory, and set the `allow_any_host` parameter to `1`, by typing the following command:

   ```
   set attr allow_any_host=1
   ```

4. Navigate to the `/subsystems/fctest1/namespaces` subdirectory, and create namespace 1 by typing the following command:

   ```
   create 1
   ```

5. Navigate to the `/subsystems/fctest1/namespaces/1` subdirectory, and set the device path to the backing storage device that you recorded previously in Step 3 in Section B.1, Configuring the elx-lpfc.conf File, by typing the following command:

   ```
   set device path=/dev/<block_device>
   ```

   <*block_device*> can be a SCSI disk or an NVMe disk.

   For example:

   ```
   set device path=/dev/sde
   ```

   Information similar to the following is displayed:

   ```
   Parameter path is now '/dev/sde'.
   ```

NOTE: This step causes the namespace to use the specified backing storage device, which already exists, as its storage media.

6. Enable namespace 1 by typing the following command:

   ```
   enable
   ```

   Information similar to the following is displayed:

   ```
   The Namespace has been enabled.
   ```

   The target NVMe over FC subsystems and namespaces have been created. You can create and enable additional namespaces by repeating Step 4 through Step 6 in this section, incrementing the namespace number each time.

7. After you have created all of the namespaces, navigate to the `/ports` directory and create a link to the desired port; in this case, Port 1, by typing the following command:

   ```
   create 1
   ```

8. Navigate to the `/ports/1` subdirectory, and set the NVMe type (the `trtype` and `adrfram` parameters) to `fc`:

   a. Type the following command:

      ```
      set addr trtype=fc
      ```

      Information similar to the following is displayed:

      ```
      Parameter trtype is now 'fc'.
      ```

   b. Type the following command:

      ```
      set addr adrfam=fc
      ```

      Information similar to the following is displayed:

      ```
      Parameter adrfam is now 'fc'.
      ```

9. Set the network address (the `traddr` parameter) to the WWNN and WWPN of the target HBA obtained in Step 1 and Step 2 in Section B.1, Configuring the elx-lpfc.conf File, by typing the following command:

   ```
   set addr traddr=nn-<WWNN1>:pn-<WWPN1>
   ```

   For example:

   ```
   set addr traddr=nn-0x20000090fa931110:pn-0x10000090fa931110
   ```

   Information similar to the following is displayed:

   ```
   Parameter traddr is now 'nn-0x20000090fa931110:pn-0x10000090fa931110'
   ```

10. Set the transport service ID (the `trsvcid` parameter) to `none` by typing the following command:

    ```
    set addr trsvcid=none
    ```

Information similar to the following is displayed:

```
Parameter trsvcid is now 'none'.
```

The port connection has been created. You can create additional port connections by repeating Step 7 through Step 10, incrementing the port number each time.

11. Link the port to the subsystem by navigating to the `/ports/1/subsystems` directory and typing the following command:

    ```
    create <subsystem>
    ```

    where <*subsystem*> is the same name as the subsystem that you created in Step 2.

    For example:

    ```
    create fctest1
    ```

12. Save the `.json` file by typing the following command:

    ```
    saveconfig <filename>.json
    ```

NOTE: If you want the configuration to be restored automatically after rebooting, this file must be named `config.json`, and it must be saved under the path `/etc/nvmet/`. Otherwise, <*filename*> can be any valid file name.

13. Exit the NVMe CLI for the target systems utility screen by typing the following command:

    ```
    exit
    ```

    The target configuration is now stored in the <*filename*>`.json` file in the directory that was current before you started the NVMeT CLI or in the `/etc/nvmet/` directory.

Unless you want to change the subsystems, namespaces, or port configurations, you perform Step 1 through Step 13 only the first time you set the configuration.

After configuring the target, you can run the `nvme_info` command to verify that your configuration is working by typing the following command on the target:

```
cat /sys/class/scsi_host/host<X>/nvme_info
```

where <*X*> is the host number.

Information similar to the following appears:

```
NVME Target Enabled  State REGISTERED
NVME Target: lpfc0 WWPN x10000090fa942779 WWNN x20000090fa942779 DID x011400

NVME Target: Statistics
LS: Rcv 00000004 Drop 00000000 Abort 00000000
LS: Xmt 00000004 Drop 00000000 Cmpl 00000004 Err 00000000
FCP: Rcv 00000002 Defer 00000000 Release 00000002 Drop 00000000
FCP Rsp: RD 00000000 rsp 00000000 WR 00000002 rsp 00000002 drop 00000000
FCP Rsp Cmpl: 00000004 err 00000000 drop 00000000
ABORT: Xmt 00000000 Cmpl 00000000
ABORT: Sol 00000000  Usol 00000000 Err 00000000 Cmpl 00000000
IO_CTX: 00001706  WAIT: cur 00000000 tot 00000000
CTX Outstanding 00000000
```

The NVMe over FC target configuration is now complete. You must now configure NVMe over FC on an initiator. The block devices configured can be discovered by the NVMe initiator system as `nvme` devices (for example: `/dev/nvme0n1`).

If you reboot the system or if you need to restore the NVMe target, type the following command:

```
nvmetcli restore <filename>.conf
```

NOTE: If a situation arises in which the initiator system boots before the target system, you must use the Linux `issue_lip` command after the target system is up. This command allows the initiator to discover the targets. On the initiator system, type the following command:

```
echo 1 > /sys/class/fc_host/host<X>/issue_lip
```

where `host<X>` represents the host number.

If you want to clear the active configuration, perform the following steps:

1. Disable the initiator links.
2. Reboot the initiator.
3. Clear the active configuration by typing the following command at the target end:
   ```
   nvmetcli clear
   ```
4. Reconfigure the target by following the instructions in Step 1 through Step 13.
5. Enable the initiator links.

You can set the saved configuration to be restored automatically after a reboot by using `systemctl` to enable the `nvmet.service`. Type the following command:

```
systemctl enable nvmet.service
```

NOTE: The service requires that the `.json` file must be named `config.json`, and it must be saved under the path `/etc/nvmet/` for it to be loaded automatically at boot time.

To disable the `nvmet.service` using `systemctl`, type the following command:

```
systemctl disable nvmet.service
```

Appendix C: Native NVMe Multipathing

Two multipathing options can be used with NVMe storage: Device Mapper (DM) and Native NVMe Multipathing.

DM is a general block-device multipathing framework that can work with devices of any protocol. DM supports the addition of hardware-specific handlers to help manage the storage device and its path management. DM has been used extensively with SCSI storage. DM can be used in its generic form with NVMe storage. However, DM currently does not support advanced NVMe features, such as ANA support. Refer to the operating system release documentation for the level of DM and NVMe storage integration supported.

Native NVMe Multipathing is a kernel-level feature that may be enabled to add an NVMe-specific multipath module to the kernel, which automatically detects subsystems and manages paths to the subsystems. The NVMe-specific module supports the NVMe ANA feature. Inclusion and enablement of the Native NVMe Multipath module is operating-system-specific. Refer to the operating system release documentation for the level of Native NVMe Multipathing supported.

Native NVMe Multipathing is included, but is not enabled by default, in SLES 12 SP4 and SLES 15. To enable Native NVMe Multipathing on SLES 12 SP4 and SLES 15 and later initiators, add `nvme-core.multipath=Y` in the kernel command line, either in the grub boot loader or during bootup.

To verify if Native NVMe Multipathing is enabled, run the following command:

```
# cat /sys/module/nvme_core/parameters/multipath
```

If Native NVMe Multipathing is enabled, the output displayed is `Y`. If Native NVMe Multipathing is not enabled, the output displayed is `N`.

When enabled, Native NVMe Multipathing changes the manner in which NVMe devices are presented and related in `/sys/class/nvme`, especially in cases where multiple paths to target namespaces exist. These changes significantly affect the views generated by the kernel and system utilities, such as `nvme-cli`. This appendix describes these new views.

When Native NVMe Multipathing is not enabled, if an NVMe device is connected to a subsystem, a controller element is created. If the subsystem is connected using multiple paths, multiple controller elements are created. Each controller element is considered a unique and separate storage entity, even though the subsystem and namespaces that the controller could access might be the same.

When Native NVMe Multipathing is enabled, controller elements are still created for each connection to a subsystem. However, when a controller is created, the subsystem is compared to the list of subsystems that have already been found in the system, and one of the following actions occurs:

- If the subsystem is not found, a new subsystem element is created and added to the system list, with the controller linked to it as a path.
- If the subsystem is found, the controller is linked to the subsystem as an additional path.

Namespaces are elements of the NVMe subsystem and not of the NVMe controller. A controller acts as a communication port to a subsystem. If a subsystem contains two controllers (communication ports), and namespace 1 is changed by controller A, controller B sees this change as an action on namespace 1 because the namespace is a singular component of the subsystem.

When Native NVMe Multipathing is not enabled, because each controller is seen as unique, namespaces for the subsystem are created for every controller on which the namespace is found. Processing of I/O requests by the device name are handled only by the controller, and they are blindly passed on to the storage device. If concurrent actions occur on namespaces on different controllers connected to the same subsystem, the application or administrator must realize that the namespace is a single entity behind the controllers, and the application or administrator must access the two controllers accordingly, so as to provide data coherency and to avoid data corruption.

When Native NVMe Multipathing is enabled, namespace device names are created when the device initially connects to the subsystem, and the controller is the first path to the subsystem. The device name reflects the subsystem of which the controller is a part. If multiple controllers connect to the same subsystem, no additional namespace device names are created because the controllers are simply paths that provide access to the same storage device. To maintain compatibility with utilities that are operating on NVMe devices on which Native NVMe Multipathing is not enabled, the name format used for the namespace device name has been kept the same. This can present some confusion because the namespace device name (for example, `/dev/nvme0n1`) contains an `nvme<`*x*`>` prefix that, when Native NVMe Multipathing is not enabled, was the controller name, but on systems on which Native NVMe Multipathing is enabled, is a prefix for the subsystem instance. In addition, for compatibility with management of NVMe controllers, the device name used for NVMe controllers (for example, `/dev/nvme2`) was left the same in systems that have Native NVMe Multipathing enabled. So, when Native NVMe Multipathing is enabled, you might see both a `/dev/nvme3` controller name and a `/dev/nvme3n1` namespace name, but no correlation exists between the `nvme3` portion of both names. In summary, when Native NVMe Multipathing is enabled, the name `/dev/nvme<`*A*`>n<`*x*`>` means *namespace number <x> on subsystem instance* <*A*>, whereas the name `/dev/nvme<`*A*`>` means *controller instance* <*A*>. The only way that the two names correlate is if controller instance <*A*> actually connects to subsystem instance <*A*> and is therefore a path to subsystem instance <*A*>.

Beyond the device name difference, the other place that this change in behavior is seen is in system utilities, such as `nvme-cli`. For example, the `nvme list` command lists all NVMe namespace devices.

NOTE: Two similar commands can be used to list all NVMe namespace devices:

- `nvme list` (with a space) is a native NVMe CLI command.
- `nvme-list-linux` (with a hyphen) is an Emulex HBA Manager CLI command.

This appendix refers to the `nvme list` command.

Thus, if two controllers are connected to a single subsystem with a single namespace, information similar to the following is displayed when Native NVMe Multipathing is not enabled:

```
Node          SN                Model   Namespace   Usage                     Format          FW Rev
/dev/nvme0n1  0000000000000000  Linux   1           268.44 GB / 268.44 GB     512 B +  0 B    4.4.131-
/dev/nvme1n1  0000000000000000  Linux   1           268.44 GB / 268.44 GB     512 B +  0 B    4.4.131-
```

On the other hand, information similar to the following is displayed in when Native NVMe Multipathing is enabled:

```
Node          SN                Model   Namespace   Usage                     Format          FW Rev
/dev/nvme0n1  0000000000000000  Linux   1           268.44 GB / 268.44 GB     512 B +  0 B    4.4.131-
```

When Native NVMe Multipathing is enabled, you can display the list of subsystems that have been detected by the system using the new `nvme list-subsys` command. For example:

```
nvme list-subsys
nvme-subsys0 - NQN=nqn.2014-08.org.nvmexpress:NVMf:uuid:19d48a6d-3644-4f65-8046-80bd88fe45a6
\
 +- nvme0 fc traddr=nn-0x20000090fa942e16:pn-0x10000090fa942e16 host_traddr=nn-0x200000109b346f1f:pn-
0x100000109b346f1f
 +- nvme3 fc traddr=nn-0x20000090fa942e16:pn-0x10000090fa942e16 host_traddr=nn-0x200000109b346f20:pn-
0x100000109b346f20
nvme-subsys1 - NQN=nqn.2014-08.org.nvmexpress:NVMf:uuid:2c6d862f-b77e-43b1-b345-7f990125088e
\
 +- nvme1 fc traddr=nn-0x20000090fa942e16:pn-0x10000090fa942e16 host_traddr=nn-0x200000109b346f1f:pn-
0x100000109b346f1f
 +- nvme6 fc traddr=nn-0x20000090fa942e16:pn-0x10000090fa942e16 host_traddr=nn-0x200000109b346f20:pn-
0x100000109b346f20
nvme-subsys2 - NQN=nqn.2014-08.org.nvmexpress:NVMf:uuid:4bbbd877-6de2-4a67-ad56-e0333e2437b5
\
 +- nvme2 fc traddr=nn-0x20000090fa942e16:pn-0x10000090fa942e16 host_traddr=nn-0x200000109b346f1f:pn-
0x100000109b346f1f
 +- nvme7 fc traddr=nn-0x20000090fa942e16:pn-0x10000090fa942e16 host_traddr=nn-0x200000109b346f20:pn-
0x100000109b346f20
nvme-subsys3 - NQN=nqn.2014-08.org.nvmexpress:NVMf:uuid:c8da7263-fefe-4563-a14c-a9f0362dcb03
\
 +- nvme5 fc traddr=nn-0x20000090fa942e16:pn-0x10000090fa942e16 host_traddr=nn-0x200000109b346f1f:pn-
0x100000109b346f1f
 +- nvme9 fc traddr=nn-0x20000090fa942e16:pn-0x10000090fa942e16 host_traddr=nn-0x200000109b346f20:pn-
0x100000109b346f20
nvme-subsys4 - NQN=nqn.2014-08.org.nvmexpress:NVMf:uuid:5f1d7788-fab1-45cf-af9c-d30dc9a9ce89
\
 +- nvme4 fc traddr=nn-0x20000090fa942e16:pn-0x10000090fa942e16 host_traddr=nn-0x200000109b346f1f:pn-
0x100000109b346f1f
 +- nvme8 fc traddr=nn-0x20000090fa942e16:pn-0x10000090fa942e16 host_traddr=nn-0x200000109b346f20:pn-
0x100000109b346f20
```

In the previous example, five subsystems are displayed. Each subsystem contains multiple controllers that represent multiple paths. Note that the `nvme-subsys` numbering is disassociated from the component controller identification, as demonstrated by `nvme-subsys3` and controllers `nvme5` and `nvme9`.

In the preceding example configuration, each of the five subsystems contain four namespaces. The `nvme list-subsys` view shows two controllers. When Native NVMe Multipathing is not enabled, each namespace and path are uniquely presented to the operating system. When Native NVMe Multipathing is enabled, each namespace is displayed once.

```
nvme list
Node            SN                  Model   Namespace   Usage                       Format          FW Rev
/dev/nvme0n1    0000000000000000    Linux   1           268.44 GB / 268.44 GB       512 B +  0 B    4.4.131-
/dev/nvme0n2    0000000000000000    Linux   1           268.44 GB / 268.44 GB       512 B +  0 B    4.4.131-
/dev/nvme0n3    0000000000000000    Linux   1           268.44 GB / 268.44 GB       512 B +  0 B    4.4.131-
/dev/nvme0n4    0000000000000000    Linux   1           268.44 GB / 268.44 GB       512 B +  0 B    4.4.131-
/dev/nvme1n1    0000000000000000    Linux   1           268.44 GB / 268.44 GB       512 B +  0 B    4.4.131-
/dev/nvme1n2    0000000000000000    Linux   1           268.44 GB / 268.44 GB       512 B +  0 B    4.4.131-
/dev/nvme1n3    0000000000000000    Linux   1           268.44 GB / 268.44 GB       512 B +  0 B    4.4.131-
/dev/nvme1n4    0000000000000000    Linux   1           268.44 GB / 268.44 GB       512 B +  0 B    4.4.131-
```

```
Node          SN                 Model   Namespace   Usage                   Format         FW Rev
/dev/nvme2n1  0000000000000000   Linux   1           268.44 GB / 268.44 GB   512 B +  0 B   4.4.131-
/dev/nvme2n2  0000000000000000   Linux   1           268.44 GB / 268.44 GB   512 B +  0 B   4.4.131-
/dev/nvme2n3  0000000000000000   Linux   1           268.44 GB / 268.44 GB   512 B +  0 B   4.4.131-
/dev/nvme2n4  0000000000000000   Linux   1           268.44 GB / 268.44 GB   512 B +  0 B   4.4.131-
/dev/nvme3n1  0000000000000000   Linux   1           268.44 GB / 268.44 GB   512 B +  0 B   4.4.131-
/dev/nvme3n2  0000000000000000   Linux   1           268.44 GB / 268.44 GB   512 B +  0 B   4.4.131-
/dev/nvme3n3  0000000000000000   Linux   1           268.44 GB / 268.44 GB   512 B +  0 B   4.4.131-
/dev/nvme3n4  0000000000000000   Linux   1           268.44 GB / 268.44 GB   512 B +  0 B   4.4.131-
/dev/nvme4n1  0000000000000000   Linux   1           268.44 GB / 268.44 GB   512 B +  0 B   4.4.131-
/dev/nvme4n2  0000000000000000   Linux   1           268.44 GB / 268.44 GB   512 B +  0 B   4.4.131-
/dev/nvme4n3  0000000000000000   Linux   1           268.44 GB / 268.44 GB   512 B +  0 B   4.4.131-
/dev/nvme4n4  0000000000000000   Linux   1           268.44 GB / 268.44 GB   512 B +  0 B   4.4.131-
/dev/nvme5n1  0000000000000000   Linux   1           268.44 GB / 268.44 GB   512 B +  0 B   4.4.131-
/dev/nvme5n2  0000000000000000   Linux   1           268.44 GB / 268.44 GB   512 B +  0 B   4.4.131-
/dev/nvme5n3  0000000000000000   Linux   1           268.44 GB / 268.44 GB   512 B +  0 B   4.4.131-
/dev/nvme5n4  0000000000000000   Linux   1           268.44 GB / 268.44 GB   512 B +  0 B   4.4.131-
```

Note that the naming includes the subsystem and namespace numbering in the form `/dev/nvme{`***`subsystem instance`***`}n{`***`namespace`***`}`. The controller (or path) is not shown.

You can also view the device nodes using the `ls` command:

```
ls -l  /dev/nvme*
0 crw------- 1 root root  10,  57 Jun 10 16:02 /dev/nvme-fabrics
0 crw------- 1 root root 246,   0 Jun 10 16:02 /dev/nvme0
0 brw-rw---- 1 root disk 259, 231 Jun 10 16:02 /dev/nvme0n1
0 brw-rw---- 1 root disk 259, 271 Jun 10 16:02 /dev/nvme0n2
0 brw-rw---- 1 root disk 259, 303 Jun 10 16:02 /dev/nvme0n3
0 brw-rw---- 1 root disk 259, 311 Jun 10 16:02 /dev/nvme0n4
0 crw------- 1 root root 246,   1 Jun 10 16:02 /dev/nvme1
0 brw-rw---- 1 root disk 259, 323 Jun 10 16:02 /dev/nvme1n1
0 brw-rw---- 1 root disk 259, 343 Jun 10 16:02 /dev/nvme1n2
0 brw-rw---- 1 root disk 259, 347 Jun 10 16:02 /dev/nvme1n3
0 brw-rw---- 1 root disk 259, 351 Jun 10 16:02 /dev/nvme1n4
0 crw------- 1 root root 246,   2 Jun 10 16:02 /dev/nvme2
0 brw-rw---- 1 root disk 259, 359 Jun 10 16:02 /dev/nvme2n1
0 brw-rw---- 1 root disk 259, 363 Jun 10 16:02 /dev/nvme2n2
0 brw-rw---- 1 root disk 259, 381 Jun 10 16:02 /dev/nvme2n3
0 brw-rw---- 1 root disk 259, 383 Jun 10 16:02 /dev/nvme2n4
0 crw------- 1 root root 246,   3 Jun 10 16:02 /dev/nvme3
0 brw-rw---- 1 root disk 259, 397 Jun 10 16:02 /dev/nvme3n1
0 brw-rw---- 1 root disk 259, 399 Jun 10 16:02 /dev/nvme3n2
0 brw-rw---- 1 root disk 259, 401 Jun 10 16:02 /dev/nvme3n3
0 brw-rw---- 1 root disk 259, 403 Jun 10 16:02 /dev/nvme3n4
0 crw------- 1 root root 246,   4 Jun 10 16:02 /dev/nvme4
0 brw-rw---- 1 root disk 259, 385 Jun 10 16:02 /dev/nvme4n1
0 brw-rw---- 1 root disk 259, 387 Jun 10 16:02 /dev/nvme4n2
0 brw-rw---- 1 root disk 259, 389 Jun 10 16:02 /dev/nvme4n3
0 brw-rw---- 1 root disk 259, 391 Jun 10 16:02 /dev/nvme4n4
0 crw------- 1 root root 246,   5 Jun 10 16:02 /dev/nvme5
0 crw------- 1 root root 246,   6 Jun 10 16:02 /dev/nvme6
```

```
0 crw------- 1 root root 246,   7 Jun 10 16:02 /dev/nvme7
0 crw------- 1 root root 246,   8 Jun 10 16:02 /dev/nvme8
0 crw------- 1 root root 246,   9 Jun 10 16:02 /dev/nvme9
```

This example demonstrates that one device node is created per namespace, with the form `/dev/nvme{`***`subsystem instance`***`}n{`***`namespace`***`}`.

Additionally, nodes are created for the controller instances `/dev/nvme{`***`controller instance`***`}`, but it is important to note that the subsystem instance and the controller instance are not correlated, as demonstrated by the previous `nvme list-subsys` example.

You can also list all of the block devices that are NVMe devices using the `ls` command. The file names are displayed as symbolic links to the actual block devices.

```
ls -l /sys/block/nvme*
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme0c222n1 -> ../devices/virtual/nvme-fabrics/ctl/
nvme0/nvme0c222n1
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme0c222n2 -> ../devices/virtual/nvme-fabrics/ctl/
nvme0/nvme0c222n2
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme0c222n3 -> ../devices/virtual/nvme-fabrics/ctl/
nvme0/nvme0c222n3
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme0c222n4 -> ../devices/virtual/nvme-fabrics/ctl/
nvme0/nvme0c222n4
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme0c227n1 -> ../devices/virtual/nvme-fabrics/ctl/
nvme3/nvme0c227n1
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme0c227n2 -> ../devices/virtual/nvme-fabrics/ctl/
nvme3/nvme0c227n2
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme0c227n3 -> ../devices/virtual/nvme-fabrics/ctl/
nvme3/nvme0c227n3
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme0c227n4 -> ../devices/virtual/nvme-fabrics/ctl/
nvme3/nvme0c227n4
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme0n1 -> ../devices/virtual/nvme-subsystem/nvme-
subsys0/nvme0n1
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme0n2 -> ../devices/virtual/nvme-subsystem/nvme-
subsys0/nvme0n2
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme0n3 -> ../devices/virtual/nvme-subsystem/nvme-
subsys0/nvme0n3
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme0n4 -> ../devices/virtual/nvme-subsystem/nvme-
subsys0/nvme0n4
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme1c223n1 -> ../devices/virtual/nvme-fabrics/ctl/
nvme1/nvme1c223n1
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme1c223n2 -> ../devices/virtual/nvme-fabrics/ctl/
nvme1/nvme1c223n2
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme1c223n3 -> ../devices/virtual/nvme-fabrics/ctl/
nvme1/nvme1c223n3
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme1c223n4 -> ../devices/virtual/nvme-fabrics/ctl/
nvme1/nvme1c223n4
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme1c229n1 -> ../devices/virtual/nvme-fabrics/ctl/
nvme6/nvme1c229n1
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme1c229n2 -> ../devices/virtual/nvme-fabrics/ctl/
nvme6/nvme1c229n2
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme1c229n3 -> ../devices/virtual/nvme-fabrics/ctl/
nvme6/nvme1c229n3
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme1c229n4 -> ../devices/virtual/nvme-fabrics/ctl/
nvme6/nvme1c229n4
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme1n1 -> ../devices/virtual/nvme-subsystem/nvme-
subsys1/nvme1n1
```

```
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme1n2 -> ../devices/virtual/nvme-subsystem/nvme-
subsys1/nvme1n2
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme1n3 -> ../devices/virtual/nvme-subsystem/nvme-
subsys1/nvme1n3
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme1n4 -> ../devices/virtual/nvme-subsystem/nvme-
subsys1/nvme1n4
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme2c224n1 -> ../devices/virtual/nvme-fabrics/ctl/
nvme2/nvme2c224n1
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme2c224n2 -> ../devices/virtual/nvme-fabrics/ctl/
nvme2/nvme2c224n2
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme2c224n3 -> ../devices/virtual/nvme-fabrics/ctl/
nvme2/nvme2c224n3
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme2c224n4 -> ../devices/virtual/nvme-fabrics/ctl/
nvme2/nvme2c224n4
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme2c230n1 -> ../devices/virtual/nvme-fabrics/ctl/
nvme7/nvme2c230n1
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme2c230n2 -> ../devices/virtual/nvme-fabrics/ctl/
nvme7/nvme2c230n2
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme2c230n3 -> ../devices/virtual/nvme-fabrics/ctl/
nvme7/nvme2c230n3
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme2c230n4 -> ../devices/virtual/nvme-fabrics/ctl/
nvme7/nvme2c230n4
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme2n1 -> ../devices/virtual/nvme-subsystem/nvme-
subsys2/nvme2n1
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme2n2 -> ../devices/virtual/nvme-subsystem/nvme-
subsys2/nvme2n2
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme2n3 -> ../devices/virtual/nvme-subsystem/nvme-
subsys2/nvme2n3
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme2n4 -> ../devices/virtual/nvme-subsystem/nvme-
subsys2/nvme2n4
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme3c228n1 -> ../devices/virtual/nvme-fabrics/ctl/
nvme5/nvme3c228n1
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme3c228n2 -> ../devices/virtual/nvme-fabrics/ctl/
nvme5/nvme3c228n2
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme3c228n3 -> ../devices/virtual/nvme-fabrics/ctl/
nvme5/nvme3c228n3
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme3c228n4 -> ../devices/virtual/nvme-fabrics/ctl/
nvme5/nvme3c228n4
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme3c232n1 -> ../devices/virtual/nvme-fabrics/ctl/
nvme9/nvme3c232n1
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme3c232n2 -> ../devices/virtual/nvme-fabrics/ctl/
nvme9/nvme3c232n2
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme3c232n3 -> ../devices/virtual/nvme-fabrics/ctl/
nvme9/nvme3c232n3
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme3c232n4 -> ../devices/virtual/nvme-fabrics/ctl/
nvme9/nvme3c232n4
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme3n1 -> ../devices/virtual/nvme-subsystem/nvme-
subsys3/nvme3n1
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme3n2 -> ../devices/virtual/nvme-subsystem/nvme-
subsys3/nvme3n2
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme3n3 -> ../devices/virtual/nvme-subsystem/nvme-
subsys3/nvme3n3
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme3n4 -> ../devices/virtual/nvme-subsystem/nvme-
subsys3/nvme3n4
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme4c226n1 -> ../devices/virtual/nvme-fabrics/ctl/
nvme4/nvme4c226n1
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme4c226n2 -> ../devices/virtual/nvme-fabrics/ctl/
nvme4/nvme4c226n2
```

```
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme4c226n3 -> ../devices/virtual/nvme-fabrics/ctl/
nvme4/nvme4c226n3
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme4c226n4 -> ../devices/virtual/nvme-fabrics/ctl/
nvme4/nvme4c226n4
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme4c231n1 -> ../devices/virtual/nvme-fabrics/ctl/
nvme8/nvme4c231n1
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme4c231n2 -> ../devices/virtual/nvme-fabrics/ctl/
nvme8/nvme4c231n2
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme4c231n3 -> ../devices/virtual/nvme-fabrics/ctl/
nvme8/nvme4c231n3
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme4c231n4 -> ../devices/virtual/nvme-fabrics/ctl/
nvme8/nvme4c231n4
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme4n1 -> ../devices/virtual/nvme-subsystem/nvme-
subsys4/nvme4n1
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme4n2 -> ../devices/virtual/nvme-subsystem/nvme-
subsys4/nvme4n2
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme4n3 -> ../devices/virtual/nvme-subsystem/nvme-
subsys4/nvme4n3
lrwxrwxrwx 1 root root 0 Jun 11 09:56 /sys/block/nvme4n4 -> ../devices/virtual/nvme-subsystem/nvme-
subsys4/nvme4n4
```

In the previous example, the symbolic links illustrate that the available namespaces are assigned a virtual controller (or path) number that is uniquely associated with the subsystem.

Appendix D: Trunking Support

Before you configure trunking on the Emulex adapter, follow the Brocade® instructions for configuring trunking on the switch.

NOTE:

- Before you enable or disable trunking on the Emulex adapter, follow the instructions from Brocade to enable or disable trunking on the switch.
- FA-PWWN and Dynamic D_Port cannot co-exist with the trunking feature on LPe35000-series adapters. If trunking is enabled, the firmware automatically disables FA-PWWN and Dynamic D_Port.
- Trunking is supported only on LPe35000-series adapters.
- Trunking is not supported at 8 Gb/s speeds, and the link will not come up at this speed.

Trunking enables you to combine multiple physical FC links to form a single logical link (aggregated port). The aggregated port's maximum link speed is the sum of the maximum link speeds of the individual physical links that comprise the aggregated port. For example, an aggregated port comprised of two physical links running at 32 Gb/s each will have a potential logical (aggregate) link speed of 64 Gb/s. The actual link speed of the aggregated port depends on the states (active/non-active) of the individual physical links that comprise the aggregated port.

The physical links that comprise an aggregated port are referred to as lanes. Both 2-lane and 4-lane aggregated ports are supported. For LPe35002 adapters, only 2-lane port aggregation is possible. If 2-lane port aggregation is configured on an LPe35002 adapter, the two physical links are combined to form a single 2-lane aggregated port whose aggregate speed is potentially 64 Gb/s, assuming that both physical links are active.

LPe35004 adapters support both 2-lane port aggregation and 4-lane port aggregation. If 2-lane port aggregation is configured on an LPe35004 adapter, the four physical links on the adapter will be divided among two separate aggregated ports. The two lowest numbered physical links will form the first aggregated port, and the two highest numbered physical links will form the second aggregated port. If 4-lane port aggregation is configured on an LPe35004 adapter, all four physical links will be combined to form a single 4-lane trunk whose aggregate speed is potentially 128 Gb/s, assuming that all 4 links are active.

There are no driver parameters to enable trunking. Trunking can be configured using the UEFI or x86 BIOS. Trunking is a one-time configuration, and the configuration is performed through the Emulex HBA Manager application. For more information on configuring trunking, refer to the *Emulex HBA Manager User Guide*.

When trunking is configured for an adapter, internal loopback and external loopback tests are supported. For more information on running loopback tests, refer to the *Emulex HBA Manager User Guide*.

You can view the existing trunk links using `sysfs`.

An example of an LPe35002 adapter, showing 2-lane port aggregation with a single SCSI host, follows:

```
# cat /sys/class/scsi_host/host14/link_state
Link Up - Ready
   Fabric
Trunk port 0: Link Up
Trunk port 1: Link Up
```

An example of an LPe35004 adapter, showing 2-lane port aggregation with a single SCSI host, follows.

```
# cat /sys/class/scsi_host/host13/link_state
Link Up - Ready
   Fabric
Trunk port 0: Link Up
Trunk port 1: Link Up
# cat /sys/class/scsi_host/host14/link_state
Link Up - Ready
   Fabric
Trunk port 2: Link Up
Trunk port 3: Link Up
```

An example of an LPe35004 adapter, showing 4-lane port aggregation with a single SCSI host, follows.

```
# cat /sys/class/scsi_host/host13/link_state
Link Up - Ready
   Fabric
Trunk port 0: Link Up
Trunk port 1: Link Up
Trunk port 2: Link Up
Trunk port 3: Link Up
```

Appendix E: FC-SP-2 Authentication (DH-CHAP) Support

NOTE:

- DH-CHAP is available only for physical ports, not for virtual ports.
- The authentication driver parameters are available only on local hosts and not for remote hosts.
- Boot from SAN is not supported when DH-CHAP authentication is enabled.
- DH-CHAP is not supported on FA-PWWN ports.
- DH-CHAP is not supported if Dynamic D_PORT is enabled.

Fibre Channel Security Protocol-2 (FC-SP-2) DH-CHAP is a secret-based authentication and key management protocol that uses the CHAP algorithm augmented with an optional Diffie-Hellmann algorithm with hash priority (MD5 and SHA1).

You can configure FC-SP-2 DH-CHAP authentication between an adapter and a switch.

FC-SP-2 DH-CHAP authentication is disabled by default. To enable DH-CHAP on Linux systems, the `lpfc_enable_auth` parameter must be passed to the driver by typing the following command:

```
insmod lpfc.ko lpfc_enable_auth=1
```

For a permanent configuration on Linux systems that will persist across system reboots, create the `/etc/modprobe.d/lpfc.conf` file, and place the following line into it:

```
options lpfc lpfc_enable_auth=1
```

NOTE: Boot from SAN is not supported when FC-SP-2 authentication is enabled.

After DH-CHAP has been activated and configured, you can manually initiate authentication per adapter, you can get, set, update, or delete a password using the Emulex HBA Manager application. For more information on manually initiating authentication, refer to the *Emulex HBA Manager User Guide*.

BROADCOM®