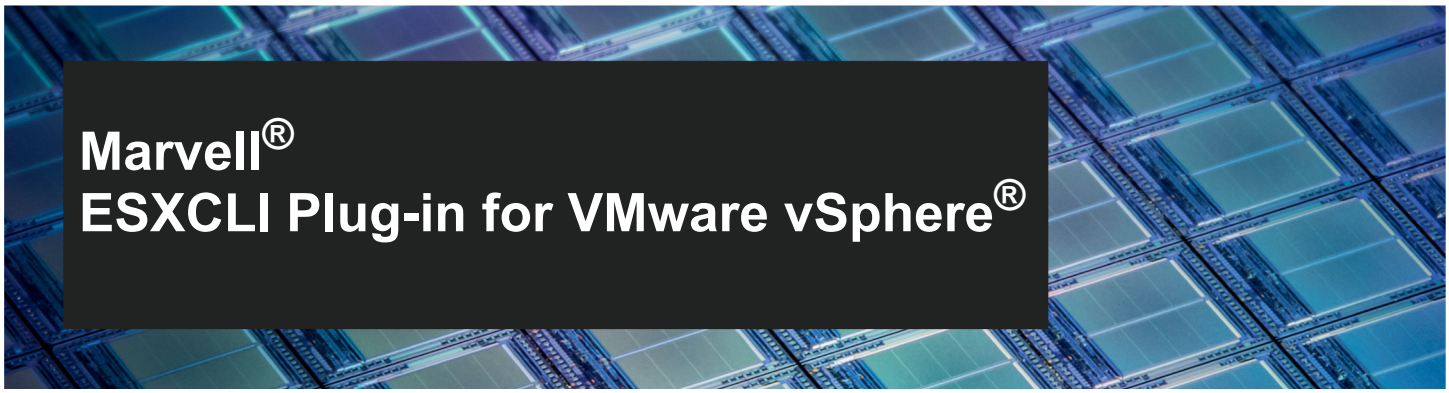

Marvell®
ESXCLI Plug-in for VMware vSphere®

2500, 2600, 2700, and 2800 Series Marvell QLogic® Fibre Channel Adapters
3400, 8400, 41000, and 45000 Series Marvell FastLinQ® Ethernet Adapters

User's Guide

Doc. No. BC0151101-00 Rev. W
March 24, 2023

Table of Contents

4 Fibre Channel CLI Command Extensions

List of Figures

List of Tables

Preface

This user's guide provides information on installing and using the Marvell® ESXCLI VMware® Plug-in. The ESXCLI plug-in extends the capabilities of the ESX® command line interface to manage the following adapters installed in VMware ESX/ESXi hosts.

- Marvell FastLinQ® 3400/8400 (578*xx*-based), 41000, and 45000 Series Ethernet Adapters
- Marvell QLogic® 2500/2600/2700/2800 Series Fibre Channel Adapters

These adapters are collectively referred to as *Marvell Adapters* throughout this guide.

Intended Audience

This guide is intended for use by administrators who are planning to deploy or have deployed one of the supported Marvell adapters in their VMware ESX/ESXi environments.

What Is in This Guide

This user's guide contains information you need to install and use the Marvell ESXCLI VMware Plug-in. This preface explains the purpose of the plug-in, identifies this guide's intended audience, describes the typographic conventions used in this guide, refers you to the license agreements, and provides technical support and contact information.

The remainder of this user's guide is organized into the following chapters:

- Chapter 1 Introduction provides the hardware and software required for using the Marvell ESXCLI VMware Plug-in.
- Chapter 2 Installation, Update, and Removal provides instructions for installing, updating, and removing the Marvell ESXCLI VMware Plug-in on VMware ESX/ESXi.
- Chapter 3 Ethernet CLI Command Extensions describes the functions of the various software components added to extend the capabilities of the Marvell ESXCLI VMware Plug-in software for Marvell Adapters.

- Chapter 4 Fibre Channel CLI Command Extensions describes the functions of the various software components added to extend the capabilities of the Marvell ESXCLI VMware Plug-in software for Marvell Adapters.
- Appendix A Revision History contains a list of changes made to this guide since the last revision.

At the end of this guide is an index to help you quickly locate the information you need.

Related Materials

For information about downloading documentation from the Marvell Web site, see "Downloading Updates and Documentation" on page xv.

Documentation Conventions

This guide uses the following documentation conventions:

- **NOTE** provides additional information.
- **CAUTION** without an alert symbol indicates the presence of a hazard that could cause damage to equipment or loss of data.
- Text in blue font indicates a hyperlink (jump) to a figure, table, or section in this guide, and links to Web sites are shown in underlined blue. For example:
  - Table 9-2 lists problems related to the user interface and remote agent.
  - See "Installation Checklist" on page 3-6.
  - For more information, visit www.marvell.com.
- Text in **bold** font indicates user interface elements such as a menu items, buttons, check boxes, or column headings. For example:
  - Click the **Start** button, point to **Programs**, point to **Accessories**, and then click **Command Prompt**.
  - Under **Notification Options**, select the **Warning Alarms** check box.
- Text in `Courier` font indicates a file name, directory path, or command line text. For example:
  - To return to the root directory from anywhere in the file structure: Type `cd /root` and press ENTER.
  - Issue the following command: **`sh ./install.bin`**

- Key names and key strokes are indicated with UPPERCASE:
  - Press CTRL+P.
  - Press the UP ARROW key.
- Text in *italics* indicates terms, emphasis, variables, or document titles. For example:
  - For a complete listing of license agreements, refer to the applicable *Software End User License Agreement*.
  - What are *shortcut keys*?
  - To enter the date type *mm/dd/yyyy* (where *mm* is the month, *dd* is the day, and *yyyy* is the year).
- Topic titles between quotation marks identify related topics either within this guide or in the online help, which is also referred to as *the help system* throughout this document.
- Command line interface (CLI) command syntax conventions include the following:
  - Plain text indicates items that you must type as shown. For example:
    - `qaucli -pr nic -ei`
  - < > (angle brackets) indicate a variable whose value you must specify. For example:
    - `<serial_number>`

    NOTE

    For CLI commands only, variable names are always indicated using angle brackets instead of *italics*.

  - `[ ]` (square brackets) indicate an optional parameter. For example:
    - `[<file_name>]` means specify a file name, or omit it to select the default file name.
  - `|` (vertical bar) indicates mutually exclusive options; select one option only. For example:
    - `on|off`
    - `1|2|3|4`

- `...` (ellipsis) indicates that the preceding item may be repeated. For example:
  - `x...` means *one* or more instances of `x`.
  - `[x...]` means *zero* or more instances of `x`.
- Vertical ellipses within command example output indicate where portions of repetitious output data have been intentionally omitted.
- `( )` (parentheses) and `{ }` (braces) are used to avoid logical ambiguity. For example:
  - `a|b c` is ambiguous
    `{(a|b) c}` means `a` or `b`, followed by `c`
    `{a|(b c)}` means either `a`, or `b c`

Technical Support

Customers should contact their authorized maintenance provider for technical support of their Marvell QLogic and FastLinQ products.

Downloading Updates and Documentation

The Marvell Web site provides periodic updates to product firmware, software, and documentation.

To download firmware, software, and documentation:

1. Go to www.marvell.com.
2. Click **Support,** and then under **Tools & Resources**, click **Driver Downloads**.
3. In the Marvell Drivers window:
   a. (MUST) Under CATEGORY, select either FIBRE CHANNEL ADAPTERS or CONVERGED NETWORK ADAPTERS.
   b. (optional) Under PLATFORM/OS, select the platform/OS that matches your system.
   c. (optional) Under PART NUMBER, select the part number for your adapter.
   d. (optional) Under KEYWORDS, type a keyword describing what you are looking for.
4. Click **Apply**.

5. Locate the firmware (boot code), software (drivers, management tools), or document (documentation for user's guides) you need, and then do one of the following:

   a. Click the blue text in the DESCRIPTION column.

   b. Click the arrow in the DOWNLOAD column.

NOTE

Marvell recommends downloading the associated Read Me and Release Notes for more information. To find them, enter either **Read Me** or **Release Notes** in the KEYWORDS search box.

A message may appear asking you to review and accept the Marvell Limited Use License Agreement.

6. If applicable, read the agreement, select the check box, and then click **I ACCEPT** to accept the end license agreement and start the download.

1 Introduction

This chapter provides the hardware and software required for using the Marvell ESXCLI VMware Plug-in.

Hardware Requirements

The Marvell ESXCLI VMware Plug-in requires a VMware ESX/ESXi server with one or more of the following installed:

- Marvell FastLinQ 3400/8400 (578*xx*-based), 41000, 45000 Series Ethernet Adapters
- Marvell QLogic 2500/2600/2700/2800 Series Fibre Channel Adapters

Software Requirements

The VMware ESX/ESXi server requires one of the following VMware vSphere® ESX/ESXi versions:

- 7.0 or later
- 8.0 or later

For remote operation, install the VMware vSphere CLI software on the local workstation. For more information about obtaining, installing, and operating the VMware software, see the VMware documentation for your system.

User Privilege Requirements

Root privileges are required on the ESX/ESXi server to install the Marvell ESXCLI VMware Plug-in, and optionally, the VMware vSphere CLI for remote plug-in operation.

2 Installation, Update, and Removal

Overview

The Marvell ESXCLI VMware Plug-in comes as a vSphere installation bundle (VIB) file. The VIB file contains the set of files and binaries required to install the provider on VMware ESX/ESXi. The `offline-bundle.zip` file contains the VIB and the necessary metadata to install the plug-in on VMware ESX/ESXi.

The Marvell ESXCLI VMware Plug-in installation has CLI stacked components, which differ for Fibre Channel and Ethernet, as shown in Figure 2-1 and described in the following paragraphs.

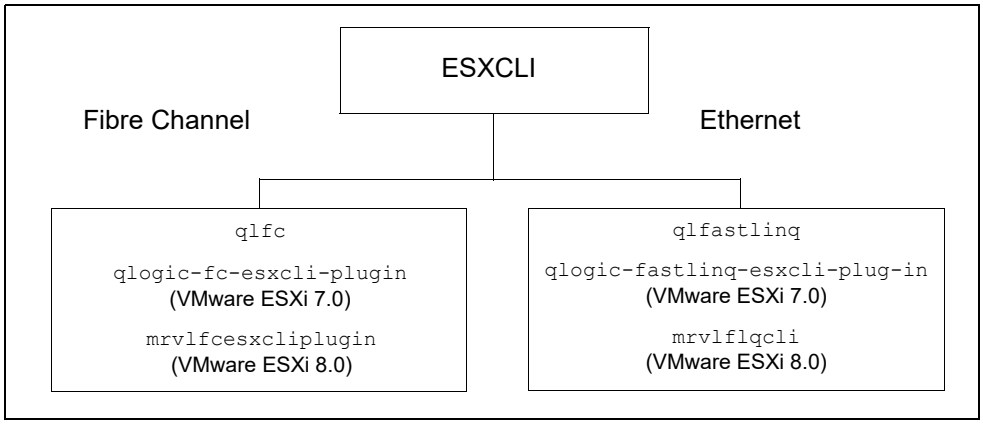

Figure 2-1. Installing Marvell ESXCLI VMware Plug-in for Fibre Channel and Ethernet

Fibre Channel

- ESXCLI Namespace: `qlfc`
  - VIB package `qlogic-fc-esxcli-plugin` (ESXi 7.0)
  - VIB package `mrvlfcesxcliplugin` (ESXi 8.0)

Ethernet

- ESXCLI Namespace: `qlfastlinq`
  - VIB package `qlogic-fastlinq-esxcli-plugin` (ESXi 7.0)
  - VIB package `mrvlflqcli` (ESXi 8.0)

Installing the Marvell ESXCLI VMware Plug-in

To install the Marvell ESXCLI VMware Plug-in:

1. Download the Marvell ESXCLI VMware Plug-in from the Marvell Web site:

   www.Marvell.com

   For instructions, see "Downloading Updates and Documentation" on page xv.

2. Copy the appropriate file into a directory on the ESXi server. Following are example file names.

   - Fibre Channel Adapters

   `MRVL-QLogic-FC-Esxcli-Plugin_<Version>-1OEM.700.1.0.15843807_<number>.zip` (VMware ESXi 7.0) or

   `MRVL-QLogic-FC-ESXCLI-Plugin_<Version>-1OEM.800.1.0.20143090_<number>.zip` (VMware 8.0)

   - Ethernet Adapters

   `qlogic-fastlinq-esxcli-plugin.zip`

3. Issue the appropriate command, where `<dir>` is the directory in which the plug-in zip file resides.

   - Fibre Channel and Ethernet Adapters

   `# esxcli software vib install -d /<path-to-package>/<package-name-zip-file>`
   (All VMware ESXi OSs) or

   `# esxcli software component apply -d /<path-to-package>/<package-name-zip-file>`
   (VMware ESXi 7.0 and later)

4. Reboot the system.

Updating the Marvell ESXCLI VMware Plug-in

To update the Marvell ESXCLI VMware Plug-in:

1. Remove the existing installation (see “Removing the Marvell ESXCLI VMware Plug-in” on page 4).
2. Download and install the new version (see Installing the Marvell ESXCLI VMware Plug-in).

Removing the Marvell ESXCLI VMware Plug-in

To remove the Marvell ESXCLI VMware Plug-in:

1. Find the name of the plug-in by issuing one of the following commands to show the VIB or component list:

   `# esxcli software vib list`
   (All VMware ESXi OSs) or

   `# esxcli software component list`
   (VMware ESXi 7.0 and later)

   The `component list` command output indicates either `MRVL-FastlinQ-Esxcli-Plugin` for Ethernet adapters or `MRVL-QLogic-FC-Esxcli-Plugin` for Fibre Channel adapters

2. Using the name that you found in Step 1 (`qlogic-fastlinq-esxcli-plugin`, `qlogic-fc-esxcli-plugin`, or `mrvlfcesxcliplugin`), issue the appropriate command to remove the Marvell ESXCLI VMware Plug-in:

   - ❑ Fibre Channel Adapters

   `# esxcli software vib remove --vibname qlogic-fc-esxcli-plugin –f`
   (VMware ESXi 7.0) or

   `# esxcli software component remove -n MRVL-QLogic-FC-Esxcli-Plugin`
   (VMware ESXi 7.0 and later) or

   `# esxcli software vib remove --vibname mrvlfcesxcliplugin -f`
   (VMware ESXi 8.0)

- Ethernet Adapters

 # esxcli software vib remove --vibname qlogic-fastlinq-esxcli-plugin –f
(VMware ESXi 7.0) or

 # esxcli software component remove -n MRVL-FastlinQ-Esxcli-Plugin
(VMware ESXi 7.0 and later) or

 # esxcli software vib remove --vibname mrvlflqcli -f
(VMware ESXi 8.0)

3 Ethernet CLI Command Extensions

This chapter describes the functions of the various software components added to extend the capabilities of the Marvell ESXCLI VMware Plug-in software for the Marvell FastLinQ Ethernet Adapters.

Sections provided in this chapter include the following:

Figure 3-1 shows the Marvell ESXCLI extension module block diagram.

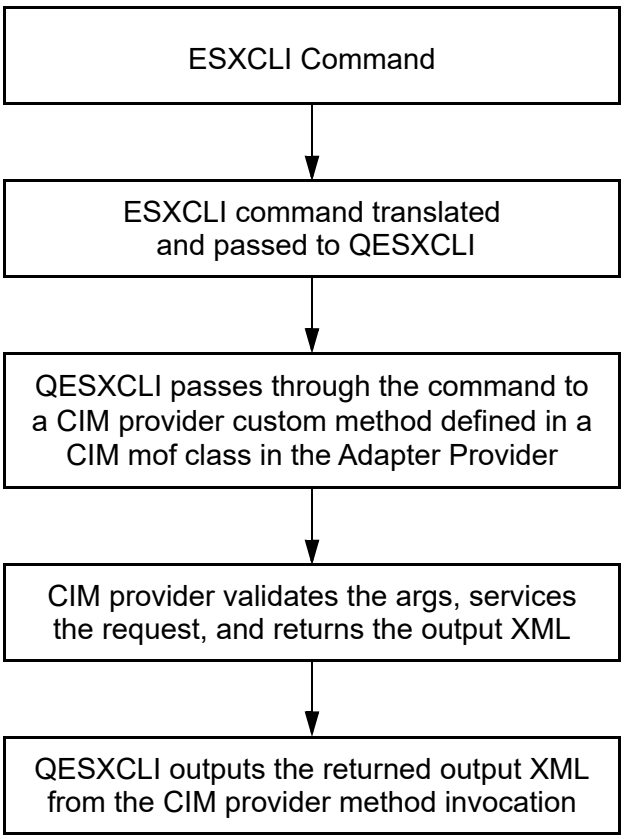

Figure 3-1. ESXCLI Extension Module Flow Chart

Extending ESXCLI Commands

The ESXCLI command line tool infrastructure allows ESXi to provide an extensible set of commands. The ESXCLI infrastructure builds each ESXCLI command using metadata that describe the input, output, and other properties for that command. Using the metadata, the infrastructure builds a set of commands that users can execute on a local or remote prompt. The Marvell ESXCLI VMware Plug-in extension is a wrapper around a standard command line executable.

The ESXCLI infrastructure presents ESXCLI commands on a local or remote prompt. Users can execute commands using predefined command line options. All Marvell ESXCLI VMware Plug-in commands are defined by metadata for input, output, and execution properties for that command. The infrastructure builds a full set of commands using the metadata for each command.

For details on how to extend ESXCLI commands, conventions, and guidelines, refer to the “Extending ESXCLI Commands Using the vCLI” section in the *VMware Host Extensions Developer’s Guide*. Access to this guide requires a VMware {code} login:

https://vdc-repo.vmware.com/vmwb-repository/dcr/915b8e00-9d0d-4ca5-ab50-de4b6ece4067/fbd6c4c0-cbb0-4e39-b97d-7e5e616b2066/HostExtensionsDevGuide.pdf?vdcDownload=1412034785_274c3924a6a76a77c987cb3ebdcaca28

Extension Command Format

The ESXCLI commands are organized as a hierarchy of *namespaces*. If you are a VMware partner, and you are creating your own ESXCLI commands, organize the commands in a consistent, logical manner to improve usability of your own commands and of ESXCLI as a whole.

Syntax

ESXCLI commands in this chapter have the following syntax:

```
esxcli [dispatcher options] <namespace1 [namespace2] ...>
<command> [input parameters]
```

- **Dispatcher options.** Predefined options for connection information such as target host, user name, and so on. Extension developers have no control over dispatcher options and do not need to be concerned about them.
- **Namespace.** Organizational structure that groups a quantity of ESXCLI objects. One namespace is required; additional namespaces are possible.

  Examples of namespaces include:

  - `nmp`
    VMware native multipathing commands.
  - `swiscsi`
    Commands in the software iSCSI namespace.
  - `corestorage`
    VMware core storage commands.
- **Command.** The command reports on or modifies the state on the system. Commands are the leaf nodes in the ESXCLI syntax. Commands are the verbs that act on the objects defined by the namespaces.
- **Input parameters.** Multiple input parameters are supported.

All ESXCLI commands in this section have examples of command usage with both a local host and a remote host. The remote host in the examples has the following credentials:

- IP address: 172.27.9.144
- Password: password
- User name: root

Format

For a local host, command syntax is as follows:

```
esxcli qlfastlinq qcc <namespace> <command> <options>
```

For a remote host, the command syntax is as follows:

```
esxcli --server <ip_address> --username <username> --password
<password> qlfastlinq qcc namespace [command] [options]
```

where `<ip_address>`, `<username>`, and `<password>` are the IP address and credentials for the remote host. The keywords `--server`, `--username`, and `--password` can be abbreviated `-s`, `-u`, and `-p` respectively.

To obtain detailed information for a specific command, type the command without arguments or with the `--help` option.

NOTE

For remote operation, the VMware vSphere CLI software must be installed on the local workstation. For more information about obtaining, installing, and operating the VMware software, see the VMware documentation for your system.

Extension Module Functional Specification

QESXCLI is the standard command line executable module invoked by the Marvell ESXCLI VMware Plug-in plug-in extension wrapper. The QESXCLI command line executable module supports multiple sets of input arguments for different commands and produces XML output for the Marvell ESXCLI VMware Plug-in extension wrapper to consume.

The non-interactive commands that are supported by QESXCLI must be published to the ESXCLI plug-in extension wrapper using an extension XML file that adheres to the metadata specification. Together, the QESXCLI and the extension XML file enable users to view and execute commands through ESXCLI (both locally as well as remotely). All of the commands are published as part of the `qlfastlinq.qcc` sub-namespace (ESXCLI namespace).

QCC Namespace-level Commands and Sub-namespaces

At the top-most level—the Marvell QConvergeConsole® (QCC) namespace level—the following commands and sub-namespaces are exposed:

```
~ # esxcli qlfastlinq qcc
```

Syntax

```
esxcli qlfastlinq qcc {cmd} [cmd options]
```

Namespaces

Available namespaces include the following:

Namespace	Description
`adapter`	QCC CLI for operations on adapters
`fcoe`	QCC CLI for operations on FCoE functions
`iscsi`	QCC CLI for operations on iSCSI functions
`nic`	QCC CLI for operations on NIC functions
`port`	QCC CLI for operations on port functions

Namespace Commands and Sub-namespaces

```
~# esxcli qlfastlinq qcc adapter
```

Syntax

```
esxcli qlfastlinq qcc adapter {cmd} [cmd options]
```

Namespaces

Available namespaces include the following:

Namespace	Description
`defCfg`	QCC CLI for adapter default configuration information
`info`	QCC CLI for adapter information
`mode`	QCC CLI for adapter mode information
`nparCfg`	QCC CLI for adapter NPAR information
`portmode`	QCC CLI for splitting the adapter in various port modes
`qinq`	QCC CLI for VLAN QinQ information
`sriov`	QCC CLI for adapter SR-IOV information
`vpd`	QCC CLI for VPD information

Available Commands

`list`
Lists identifiers for the Marvell Adapters discovered on the host. See List Adapters.

`update`
Updates the flash on a Marvell Adapter. See Update Adapters.

List Adapters

To list all supported 3400/8400 Series and 41000/45000 Series Adapters, issue the `list` command as follows on the local host:

```
~ # esxcli qlfastlinq qcc adapter list
```

The following example shows the `list` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc adapter
list
```

The output for both commands is:

```
Adapters
------------
Adapter_17_0
Adapter_27_0
```

The output contains the PCI bus and device number in decimal format as part of the adapter identifier.

Update Adapters

To update the flash on a Marvell Adapter, issue the `adapter update` command as follows:

```
~ # esxcli qlfastlinq qcc adapter update
```

Syntax

```
esxcli qlfastlinq qcc adapter update [cmd options]
```

Keywords

`-f|--flashfile <str>` = Flash file (required)

`-i|--identifier <str>` = Adapter identifier (required). Obtain the identifier from the adapter `list` output.

Example

The following example shows the `adapter update` command on a local host:

```
~ # esxcli qlfastlinq qcc adapter update -i Adapter_10_0 -f
/scratch/dell_ah_100020.bin
```

The flash file must be on the remote host before issuing the `adapter update` command as described in the following steps.

1. Determine the name of the volume on the remote host in which to store the flash file. You will use the volume name and VMFS UUID in subsequent steps.

```
# esxcli -s 172.27.9.144 -u root -p password storage vmfs extent list

Volume Name     VMFS UUID                            Extent Number  Device Name
--------------  -----------------------------------   -------------  -----------
datastore1 (3)  4d435b11-2d3d0b17-a566-000af73e458c                 0
```

2. Create a temporary directory on the remote host in which to store the flash file using the name of the volume that you obtained in Step 1. The following command creates a directory called `tmp` on the volume `datastore1 (3)`. You will specify this directory when you update the flash in Step 4.

```
# vifs.pl --server 172.27.9.144 --username root --password password
--mkdir "[datastore1 (3)] tmp"
```

3. Upload the flash file to the remote host.

```
# vifs.pl --server 172.27.9.144 --username root --password password
--put ql_mbi_8712.bin "[datastore1 (3)] tmp/ql_mbi_8712.bin"
```

4. Update the adapter flash on the remote host. The flash file path is indicated in `/vmfs/volumes/<vmfs_uuid>/tmp/ql_mbi_8712.bin`, where the value for `<vmfs_uuid>` comes from Step 1.

The following example shows the `adapter update` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc adapter
update -i Adapter_10_0 -f /vmfs/volumes/
4d435b11-2d3d0b17-a566-000af73e458c/tmp/dell_ah_100020.bin
```

The output for both commands is:

```
Message: Success
Reboot Required: true
```

Adapter Information

To obtain information about a specific Marvell Adapter, issue the `info` command as follows:

```
~ # esxcli qlfastlinq qcc adapter info
```

Syntax

```
esxcli qlfastlinq qcc adapter info {cmd} [cmd options]
```

Available Commands

`get`
Shows information about the adapter. See Get Adapter Information.

Get Adapter Information

To obtain information about a Marvell Adapter, issue the `get` command as follows:

```
~# esxcli qlfastlinq qcc adapter info get
```

Syntax

```
esxcli qlfastlinq qcc adapter info get [cmd options]
```

Keywords

`-i|--identifier <str>` = Adapter identifier (required). Obtain the identifier from the adapter `list` output.

Example

The following example shows the `adapter info get` command on a local host:

```
~# esxcli qlfastlinq qcc adapter info get -i Adapter_17_0
```

The following example shows the `adapter info get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc adapter
info get -i Adapter_17_0
```

The output for both commands is:

```
Adapter Information:
   General Parameters:
         Asic Version: QL41401S A2
         Bus Width: PCI-E (8X)
         Description: HPE StoreFabric CN1300R 10/25Gb CNA
         Device Id: 0x8070
         Fw Versions: Bootcode Version:8.37.6.0,
         PXE Boot Version:PCI30 MBA 2.0.18;EFI x64 4.1.5.6,
         MBI Version:8.37.7
         Manufacturer: QLogic Corporation
         Port Mode: NA
         Serial Number: REE1740B20939
         Subsystem Device Id: 0x21b
         Subsystem Vendor Id: 0x1590
         Temperature In Celsius: 63
         Title: HPE StoreFabric CN1300R 10/25Gb CNA
         Vendor Id: 0x1077
   Message: Success
```

Adapter Mode

To display and set the adapter's current mode (Single-Function or Multi-Function), issue the `adapter mode` namespace command as follows:

```
~# esxcli qlfastlinq qcc adapter mode
```

Syntax

```
esxcli qlfastlinq qcc adapter mode {cmd} [cmd options]
```

Available Commands

`get`
Shows mode information for the specified Marvell Adapter. See Get Adapter Mode.

`set`
Sets the mode for the specified Marvell Adapter. See Set Adapter Mode.

Get Adapter Mode

To read the mode of a Marvell Adapter, issue the `adapter mode get` command as follows:

```
~# esxcli qlfastlinq qcc adapter mode get
```

Syntax

```
esxcli qlfastlinq qcc adapter mode get [cmd options]
```

Keywords

`-i|--identifier <str>` = Adapter identifier (required). Obtain the identifier from the adapter `list` output.

Example

The following example shows the `adapter mode get` command on a local host:

```
~# esxcli qlfastlinq qcc adapter mode get -i Adapter_17_0
```

The following example shows the `adapter mode get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc adapter
mode get -i Adapter_17_0
```

The output for both commands is:

```
Message: Success
   Mode: Single-Function
```

Set Adapter Mode

To set the mode of a Marvell Adapter to Single-Function or Multi-Function, issue the `adapter mode set` command as follows:

```
~# esxcli qlfastlinq qcc adapter mode set
```

Syntax

```
esxcli qlfastlinq qcc adapter mode set [cmd options]
```

Keywords

`-e|--ep-mode<str>` = EP mode (Dell® 41000/45000 Series adapters only)
`-i|--identifier<str>` = Adapter identifier (required)
`-m|--mode<str>` = Adapter mode (required)

Parameter	Value (strings are case-sensitive)
`identifier`	Adapter identifier from the adapter `list` command
`mode`	Single-Function, Multi-Function, or UFP. UFP mode is supported on selected IBM and Lenovo 3400/8400 Series Adapters.
`ep-mode`	Enabled, Disabled (for Dell 41000/45000 Series Adapters only)

Example

The following example shows the `adapter mode set` command on a local host:

```
~# esxcli qlfastlinq qcc adapter mode set -i Adapter_17_0 -m
Single-Function
```

The following example shows the `adapter mode set` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc adapter
mode set -i Adapter_17_0 -m Single-Function
```

The output for both commands is:

```
Message: Success
   Mode: Single-Function
   Reboot Required: true
```

The command output shows the new setting, which will be active after reboot.

Adapter Default Configuration

To display adapter data and change adapter protocol settings when the mode is set to Single-Function, issue the `adapter defCfg` namespace command:

```
~# esxcli qlfastlinq qcc adapter defCfg
```

Syntax

```
esxcli qlfastlinq qcc adapter defCfg {cmd} [cmd options]
```

Available Commands

`get`
Shows default configuration information for the specified Marvell Adapter. See Get Default Configuration.

`setFile`
Sets default configuration information for the specified Marvell Adapter in a file. See Set Default Configuration File.

`set`
Sets default configuration information for the specified Marvell Adapter using a cached file. See Set Default Configuration.

Get Default Configuration

To view default configuration data for a Marvell Adapter that is in Single-Function mode (Multi-Function mode is not supported), issue the `defCfg get` command as follows:

```
~# esxcli qlfastlinq qcc adapter defCfg get
```

Syntax

```
esxcli qlfastlinq qcc adapter defCfg get [cmd options]
```

Keywords

`-i|--identifier<str>` = Adapter identifier (required). Obtain the identifier from the adapter `list` output.

Example

The following example shows the `adapter defCfg get` command on a local host:

```
~# esxcli qlfastlinq qcc adapter defCfg get -i Adapter_17_0
```

The following example shows the `adapter defCfg get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc adapter
defCfg get -i Adapter_17_0
```

The output for both commands is:

```
Default Configuration:
   Default Config:
         Port Config:
         Ethernet: Enabled
         Fcoe: Enabled
         Iscsi: Enabled
         Iwarp: NA
         Port Number: 0
         Roce: Disabled

         Ethernet: Enabled
```

```
            Fcoe: Enabled
            Iscsi: Enabled
            Iwarp: NA
            Port Number: 1
            Roce: Disabled
      Message: Supported Mode
      Mode: Single-Function
```

Set Default Configuration File

To save user-provided data in a temporary cache file for a Marvell Adapter that is in Single-Function mode (Multi-Function mode is not supported), issue the `adapter defCfg setFile` command as follows:

```
~# esxcli qlfastlinq qcc adapter defCfg setFile
```

The `setFile` command creates a cache file of the default configuration on the test system, which is used by the `set` command to save the setting in the flash.

Syntax

```
esxcli qlfastlinq qcc adapter defCfg setFile [cmd options]
```

Keywords

`-f|--fcoe <str>` = FCoE

`-i|--identifier <str>` = Adapter identifier (required).

`-c|--iscsi <str>` = iSCSI

`-w|--iwarp <str>` = Internet wide area RDMA protocol (iWARP) (for 41000/45000 Series adapters only; currently not supported by VMware)

`-r|--roce <str>` = RDMA over Converged Ethernet (RoCE) (for 41000/45000 Series adapters only)

`-p|--port-number <long>` = Port number (required)

Parameter	Value (strings are case-sensitive)
`identifier`	Adapter identifier from the adapter `list` command
`port-number`	Physical port number of the adapter: `0` for Single port `0-1` for Dual port `0-3` for Quad-Port

Parameter	Value (strings are case-sensitive)
`fcoe`	Enabled, Disabled
`iscsi`	Enabled, Disabled
`roce`	Enabled, Disabled
`iwarp`	Enabled, Disabled

Example

The following example shows the `adapter defCfg SetFile` command on a local host:

```
~# esxcli qlfastlinq qcc adapter defCfg setFile -i Adapter_17_0 -p
0 -f Enabled
```

The following example shows the `adapter defCfg SetFile` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc adapter
defCfg setFile -i Adapter_17_0 -p 0 -f Enabled
```

The output for both commands is:

```
Default Configuration:
   Default Config:
         Port Config:
         Ethernet: Enabled
         Fcoe: Enabled
         Iscsi: Enabled
         Iwarp: NA
         Port Number: 0
         Roce: Disabled

         Ethernet: Enabled
         Fcoe: Enabled
         Iscsi: Enabled
         Iwarp: NA
         Port Number: 1
         Roce: Disabled
Message: Supported Mode
   Mode: Single-Function
```

The command output includes a combination of the existing settings and the user-modified settings saved in the cache file.

Set Default Configuration

The `adapter defCfg set` command sets user-provided or default configuration data from a temporary cache file to the flash when the adapter is in Single-Function mode (not supported in Multi-Function mode). This command fails if the cache file is not present. On successful completion, the `defCfg set` command cache file is deleted.

```
~# esxcli qlfastlinq qcc adapter defCfg set
```

Syntax

```
esxcli qlfastlinq qcc adapter defCfg set [cmd options]
```

Keywords

`-i|--identifier <str>` = Adapter identifier (required). Obtain the identifier from the adapter `list` output.

Example

The following example shows the `adapter defCfg set` command on a local host:

```
~# esxcli qlfastlinq qcc adapter defCfg set -i Adapter_17_0
```

The following example shows the `adapter defCfg set` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc adapter
defCfg set -i Adapter_17_0
```

The output for both commands is:

```
Default Configuration:
   Default Config:
         Port Config:
         Ethernet: Enabled
         Fcoe: Disabled
         Iscsi: Enabled
         Iwarp: NA
         Port Number: 0
         Roce: Enabled

         Ethernet: Enabled
         Fcoe: Enabled
         Iscsi: Enabled
         Iwarp: NA
         Port Number: 1
         Roce: Enabled
```

```
Message: Supported Mode
Mode: Single-Function
Reboot Required: true
```

The output shows the new setting, which becomes active after a reboot.

Adapter NPAR Configuration

To display adapter data and change adapter function settings when the mode is set to Multi-Function (not supported in Single-Function mode), issue the `adapter nparCfg` namespace command:

```
~# esxcli qlfastlinq qcc adapter nparCfg
```

Syntax

```
esxcli qlfastlinq qcc adapter nparCfg {cmd} [cmd options]
```

Available Commands

`get`
Shows NPAR information for a specific Marvell Adapter. See Get NPAR Configuration.

`setportcfg`
Sets NPAR port level configuration information for a specific Marvell Adapter in a file. See Set NPAR Port Configuration.

`setfunccfg`
Sets NPAR function level configuration information for a specific Marvell Adapter in a file. See Set NPAR Function Configuration.

`set`
Sets NPAR configuration for a specific Marvell Adapter. See Set NPAR Configuration.

Get NPAR Configuration

The `adapter nparCfg get` command displays data when the adapter is in Multi-Function mode (not supported in Single-Function mode). To get NPAR information about the specified Marvell Adapter, issue the following command:

```
~# esxcli qlfastlinq qcc adapter nparCfg get
```

Syntax

```
esxcli qlfastlinq qcc adapter nparCfg get [cmd options]
```

Keywords

`-i|--identifier <str>` = Adapter identifier (required). Obtain the identifier from the adapter `list` output.

Example

The following example shows the `adapter nparCfg get` command on a local host:

```
~# esxcli qlfastlinq qcc adapter nparCfg get -i Adapter_16_0
```

The following example shows the `adapter nparCfg get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc adapter
nparCfg get -i Adapter_16_0
```

The output for both commands is:

```
Npar Configuration:
   Message: Supported Mode
   Mode: Multi Function
   NPAR:
         EPMode: Disabled
         Functions:
         Ethernet: Enabled
         Fcoe: Disabled
         Function Number: 0
         Iscsi: Disabled
         Iwarp: Disabled
         Maximum Bandwidth: 100
         Relative Bandwidth Weight: 0
         Roce: Disabled

         Ethernet: Enabled
         Fcoe: Disabled
         Function Number: 1
         Iscsi: Disabled
         Iwarp: Disabled
         Maximum Bandwidth: 100
         Relative Bandwidth Weight: 0
         Roce: Disabled
.
.
```

.

```
        Ethernet: Enabled
        Fcoe: Disabled
        Function Number: 15
        Iscsi: Disabled
        Iwarp: Disabled
        Maximum Bandwidth: 100
        Relative Bandwidth Weight: 0
        Roce: Disabled
        Ports:
        Flow Control: Rx/Tx
        Port Number: 0

        Flow Control: Rx/Tx
        Port Number: 1
```

Set NPAR Port Configuration

The `adapter nparCfg setportcfg` command saves user-provided port-level data in a temporary cache file when the adapter is in Multi-Function mode (not supported in Single-Function mode; that is, the system must be in NPAR mode). This cache file is used by the `set` command to save settings in the flash. To set NPAR port configuration information for a specified Marvell Adapter in a file, issue the following command:

```
~# esxcli qlfastlinq qcc adapter nparCfg setportcfg
```

Syntax

```
esxcli qlfastlinq qcc adapter nparCfg setportcfg [cmd options]
```

Keywords

`-f|--flow-control <str>` = Flow control

`-i|--identifier <str>` = Adapter identifier (required)

`-p|--port-number <long>` = Port number (required)

Parameter	Value (strings are case-sensitive)
identifier	Adapter identifier from the adapter list command
port-number	Physical port number of the adapter: 0 for Single port 0-1 for Dual port 0-3 for Quad port
flow-control	Auto, Tx, Rx, Rx/Tx, Disabled

Example

The following example shows the adapter nparCfg setportcfg command on a local host. This example sets the flow control value for port **0** of the adapter named Adapter_16_0.

```
~# esxcli qlfastlinq qcc adapter nparCfg setportcfg -i
Adapter_16_0 -p 0 -f Rx/Tx
```

The following example shows the adapter nparCfg setportcfg command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc adapter
nparCfg setportcfg -i Adapter_16_0 -p 0 -f Rx/Tx
```

The output for both commands is:

```
Npar Configuration:
   Message: Supported Mode
   Mode: Multi Function
   NPAR:
         EPMode: Disabled
         Functions:
         Ethernet: Enabled
         Fcoe: Disabled
         Function Number: 0
         Iscsi: Disabled
         Iwarp: Disabled
         Maximum Bandwidth: 100
         Relative Bandwidth Weight: 0
         Roce: Disabled

         Ethernet: Enabled
         Fcoe: Disabled
         Function Number: 1
         Iscsi: Disabled
         Iwarp: Disabled
```

```
            Maximum Bandwidth: 100
            Relative Bandwidth Weight: 0
            Roce: Disabled
.
.
.
            Ethernet: Enabled
            Fcoe: Disabled
            Function Number: 15
            Iscsi: Disabled
            Iwarp: Disabled
            Maximum Bandwidth: 100
            Relative Bandwidth Weight: 0
            Roce: Disabled
            Ports:
            Flow Control: Rx/Tx
            Port Number: 0

            Flow Control: Rx/Tx
            Port Number: 1
```

The command output is a combination of existing settings and the user-modified settings saved in the cache file.

Set NPAR Function Configuration

The `adapter nparCfg setfunccfg` command saves user-provided function-level data in a temporary cache file when the adapter is in Multi-Function mode (not supported in Single-Function mode). This cache file is used by the `set` command to save settings in the flash. To set NPAR function configuration information on a specified Marvell Adapter in a file, issue the following command:

```
~# esxcli qlfastlinq qcc adapter nparCfg setfunccfg
```

Syntax

```
esxcli qlfastlinq qcc adapter nparCfg setfunccfg [cmd options]
```

Keywords

`-e|--ethernet <str>` = Ethernet

`-f|--fcoe <str>` = FCoE

`-n|--function-number <long>` = Function number (required)

`-i|--identifier <str>` = Adapter identifier (required)

`-c|--iscsi <str>` = iSCSI

`-p|--iwarp <str>` = iWARP (for 41000/45000 Series adapters)
`-r|--roce <str>` = RDMA over Converged Ethernet (RoCE) (for 41000/45000 Series adapters)
`-w|--max-bw <long>` = Maximum bandwidth
`-m|--min-bw <long>` = Minimum bandwidth

Parameter	Value (strings are case-sensitive)
`identifier`	Adapter identifier from the adapter `list` command
`function-number`	Function number of the adapter: `0-7` for 3400/8400 Series adapters. For more information, see Note 1. `0-15` for 41000/45000 Series adapters. For more information, see Note 2.
`ethernet`	Enabled, Disabled
`iscsi`	Enabled, Disabled
`iwarp`	Enabled, Disabled.
`fcoe`	Enabled, Disabled
`roce`	Enabled, Disabled
`max-bw`	Maximum bandwidth (`0-100`)
`min-bw`	Minimum bandwidth (`0-100`). For more information, see Note 3.

Note 1

The 3400/8400/57800-based adapters support up to 8 functions in NPAR mode (except for the single port 3441/57811-based adapters and quad port 57800-based adapters). These adapters do not support 16 functions in NPAReP mode.

The dual 1GbE and dual 10GbE port 57800-based adapters support functions `0-5`, but the 1GbE partitions (functions `2-3`) do not support the NPAR feature. The single port 3441/57811-based adapters support functions `0-3`.

See also footnotes 1 and 2.

Note 2

When the `Function-number` value is `0–15` (41000/45000 Series adapters):

- The 41000/45000 adapters support up to 8 functions in NPAR[1] mode; and up to 16 functions in NPAReP[2] mode.
- VMware ESXi 7.0 and later supports up to 16 physical (functions `0–15`) per adapter.
- The single/dual/quad-port 41000 Series adapters[1] and dual/quad-port 45000 Series adapters[1] support values `0–7`.
- The single-port 45000 Series adapters[2] support values `0`, `2`, `4`, `6`, `8`, `10`, `12`, `14`.
- The single-port 45000 Series adapters[1] support values `0`, `2`, `4`, `6`.
- The dual 1G-port and dual 10G-port based adapters[1] support values `0`, `1`, `4`, `5`. Functions `2` and `3` are the two fixed speed 1GbE RJ45 ports (which do not support the any NPAR features), and therefore are not valid values. Functions `6` and `7` are not used.
- The dual 1G-port and dual 10G-port based adapters[2] support values `0`, `1`, `4`, `5`, `8`, `9`, `12`, `13`; the two fixed-speed 1G RJ45 ports (which use values `2` and `3`) do not support the NPAReP feature. Functions `6`, `7`, `10`, `11`, `14`, `15` are not used.

Note 3

The sum total of all partitions of a single port's relative bandwidth weight (`min-bw`) value should be either `0` (all partitions are set to 0) or `100`.

Example

The following example shows the `adapter nparCfg setfunccfg` command on a local host:

```
~# esxcli qlfastlinq qcc adapter nparCfg setfunccfg -i
Adapter_16_0 -n 2 -m 10
```

The following example shows the `adapter nparCfg setfunccfg` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc adapter
nparCfg setfunccfg -i Adapter_16_0 -n 2 -m 10
```

[1] If the server does not support PCI Alternative-Routing ID Interpretation (ARI), or the adapter is set to 8-partition NPAR mode instead of 16-partition NIC extended partitioning (NPAReP) mode.

[2] If the PCI ARI-supported server is set to 16-partition NPAReP mode.

The output for both commands is:

```
Npar Configuration:
   Message: Supported Mode
   Mode: Multi Function
   NPAR:
         EPMode: Disabled
         Functions:
         Ethernet: Enabled
         Fcoe: Disabled
         Function Number: 0
         Iscsi: Disabled
         Iwarp: Disabled
         Maximum Bandwidth: 100
         Relative Bandwidth Weight: 0
         Roce: Disabled

         Ethernet: Enabled
         Fcoe: Disabled
         Function Number: 1
         Iscsi: Disabled
         Iwarp: Disabled
         Maximum Bandwidth: 100
         Relative Bandwidth Weight: 0
         Roce: Disabled
.
.
.
         Ethernet: Enabled
         Fcoe: Disabled
         Function Number: 15
         Iscsi: Disabled
         Iwarp: Disabled
         Maximum Bandwidth: 100
         Relative Bandwidth Weight: 0
         Roce: Disabled
         Ports:
         Flow Control: Rx/Tx
         Port Number: 0
```


Doc. No. BC0151101-00 Rev. W

```
        Flow Control: Rx/Tx
        Port Number: 1
```

The output shows a combination of the existing settings and the user-modified settings that are saved in the cache file.

Set NPAR Configuration

The `adapter nparCfg set` command imports user-provided data from a temporary cache file to the flash when the adapter is in Multi-Function mode (not supported for Single-Function mode). This command fails if the cache file is not present. On successful completion of the `adapter nparCfg set` command, the cache file is deleted.

To import NPAR information for a specified Marvell Adapter, issue the following command:

```
~# esxcli qlfastlinq qcc adapter nparCfg set
```

Syntax

```
esxcli qlfastlinq qcc adapter nparCfg set [cmd options]
```

Keywords:

`-i|--identifier <long>` = Adapter identifier (required). Obtain the identifier from the adapter `list` output

Example

The following example shows the `adapter nparCfg set` command on a local host:

```
~# esxcli qlfastlinq qcc adapter nparCfg set -i Adapter_16_0
```

The following example shows the `adapter nparCfg set` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc adapter
nparCfg set -i Adapter_16_0
```

The output for both commands is:

```
Npar Configuration:
   Message: Supported Mode
   Mode: Multi Function
   NPAR:
         EPMode: Disabled
         Functions:
         Ethernet: Enabled
         Fcoe: Disabled
         Function Number: 0
```

```
            Iscsi: Disabled
            Iwarp: NA
            Maximum Bandwidth: 100
            Relative Bandwidth Weight: 0
            Roce: NA

            Ethernet: Enabled
            Fcoe: Disabled
            Function Number: 1
            Iscsi: Disabled
            Iwarp: NA
            Maximum Bandwidth: 100
            Relative Bandwidth Weight: 0
            Roce: NA
.
.
.
            Ethernet: Enabled
            Fcoe: Disabled
            Function Number: 15
            Iscsi: Disabled
            Iwarp: NA
            Maximum Bandwidth: 100
            Relative Bandwidth Weight: 0
            Roce: NA
            Ports:
            Flow Control: Rx/Tx
            Port Number: 0

            Flow Control: Rx/Tx
            Port Number: 1
   Reboot Required: true
```

The output shows the new settings, which become active after a reboot if the command is successful.

Adapter SR-IOV Configuration

The SR-IOV configuration commands display SR-IOV data when the adapter is in Single-Function or Multi-Function mode (that is, SR-IOV over NPAR/NPAReP is supported on the 3400/8400/578*xx*-based Series adapters and 41000/45000 Series adapters).

```
~# esxcli qlfastlinq qcc adapter sriov
```

Syntax

```
esxcli qlfastlinq qcc adapter sriov {cmd} [cmd options]
```

Available Commands

`get`
Shows SR-IOV information for the specified Marvell Adapter. See Get SR-IOV Information.

`setFile`
Saves SR-IOV information for the specified Marvell Adapter in a cache file. See Set SR-IOV File.

`set`
Saves SR-IOV information for the specified Marvell Adapter. See Set SR-IOV.

Get SR-IOV Information

To view SR-IOV data read from flash for a Marvell Adapter, issue the following command:

```
~# esxcli qlfastlinq qcc adapter sriov get
```

Syntax

```
esxcli qlfastlinq qcc adapter sriov get [cmd options]
```

Keywords

`-i|--identifier <long>` = Adapter identifier (required). Obtain the identifier from the adapter `list` output

Example

The following example shows the `adapter sriov get` command on a local host:

```
~# esxcli qlfastlinq qcc adapter sriov get -i Adapter_17_0
```

The following example shows the `adapter sriov get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc adapter
sriov get -i Adapter_17_0
```

The output for both commands is:

```
SRIOV Configuration:
   Message: Success
   Mode: Single-Function
   SRIOV: Enabled
   Sriov Config:
         Pf Number: 0
         VFs Per PF: 24

         Pf Number: 1
         VFs Per PF: 32
```

Set SR-IOV File

The `adapter sriov setFile` command saves user-provided SR-IOV data for a specified Marvell Adapter in a temporary cache file. This cache file is used by the `set` command to save settings in the flash. To set the cache file, issue the following command:

```
~# esxcli qlfastlinq qcc adapter sriov setFile
```

Syntax

```
esxcli qlfastlinq qcc adapter sriov setFile [cmd options]
```

Keywords

`-i|--identifier <str>` = Adapter identifier (required)

`-v|--number-of-vfs <long>` = Number of virtual functions

`-n|--pf-number <long>` = Physical function number (required)

`-o|--sriov <str>` = SR-IOV

Parameter	Value (strings are case-sensitive)
`identifier`	Adapter identifier from the adapter `list` command
`pf-number`	Adapter's Ethernet physical function port number: `0` for a single port `0-1` for a dual port `0-3` for a quad port in Single-Function mode `0-7` for 3400/8400 Series adapters `0-15` for 41000/45000 Series adapters in Multi-Function mode

Parameter	Value (strings are case-sensitive)
`sriov`	Enabled, Disabled
`number-of-vfs`	Integer value of modulo 8. No VFs can be assigned to a physical function enabled with a storage personality (iSCSI offload or FCoE-offload). Maximum values: `128` for a 3400/8400 Series adapter. For more information, see Notes. `192` for a 41000 Series adapter. For more information, see Notes. `240` for a 45000 Series adapter. For more information, see Notes.

Notes

When the `Number-of-vfs` parameter is `128`:

- `64` per port on single and dual port adapter
- `32` per port on quad port adapters.

In addition, VFs can be user-distributed across Ethernet physical functions on the same port.

When the `Number-of-vfs` parameter is `192` (for a 41000 Series adapter), the actual number of VFs is hard divided across Ethernet physical functions of an adapter–the VFs are not user re-distributable. For single-port adapters, there are 64 VFs (not 192) on the *default* mode single Ethernet physical function per port; in NPAReP mode, there are 16 VFs on the first eight Ethernet physical functions and 8 VFs on the last eight Ethernet physical functions. For dual-port adapters, there are 64 VFs on the *default* mode single Ethernet physical function per port; in NPAReP mode, there are 16 VFs on the first eight Ethernet physical functions and 8 VFs on the last eight Ethernet physical functions. For quad-port adapters, there are 48 VFs on the *default* mode single Ethernet physical function per port; in NPAReP mode, there are 16 VFs on the first eight Ethernet physical functions and 8 VFs on the last eight Ethernet physical functions.

When the `Number-of-vfs` parameter is `240` (for a 45000 Series adapter), the actual number of VFs is hard divided across Ethernet physical functions of an adapter–the VFs are not user re-distributable. For single-port adapters, there are 64 VFs (not 120) on the *default* mode single Ethernet physical function per port; in NPAReP mode, there are 16 VFs on the first seven Ethernet physical functions and 8 VFs on the last Ethernet physical function. For dual-port adapters, there are 64 VFs on the *default* mode single Ethernet physical function per port; in NPAReP mode, there are 16 VFs on the first fourteen Ethernet physical functions and 8 VFs on the last two Ethernet physical functions. For quad-port adapters, there are 64 VFs on the *default* mode single Ethernet physical function per port; in NPAReP mode, there are 16 VFs on the first fourteen Ethernet physical functions and 8 VFs on the last two Ethernet physical functions.

Example

The following example shows the `adapter sriov setFile` command on a local host:

```
~ # esxcli qlfastlinq qcc adapter sriov setFile -i Adapter_17_0 -n
1 -v 32
```

The following example shows the `adapter sriov setFile` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc adapter
sriov setFile -i Adapter_17_0 -n 1 -v 32
```

The output for both commands is:

```
SRIOV Configuration:
   Message: Success
   Mode: Single-Function
   SRIOV: Enabled
   Sriov Config:
         Pf Number: 0
         VFs Per PF: 24

         Pf Number: 1
         VFs Per PF: 32
```

The output is a combination of existing settings and user-modified settings saved in the cache file.

Set SR-IOV

The `adapter sriov set` command imports user-provided SR-IOV information for a specified Marvell Adapter from a temporary cache file to the flash. This command fails if the cache file is not present. On successful completion of the set command, the cache file is deleted.

```
~# esxcli qlfastlinq qcc adapter sriov set
```

Syntax

```
esxcli qlfastlinq qcc adapter sriov set [cmd options]
```

Keywords

`-i|--identifier <str>` = Adapter identifier (required). Obtain the identifier from the adapter `list` output.

Parameter	Value (strings are case-sensitive)
`identifier`	Adapter identifier from the adapter `list` command
`pf-number`	Adapter's physical function port number: `0` for a single port `0-1` for a dual port `0-3` for a quad port in Single-Function mode `0-7` for 3400/8400 Series adapters `0-15` for 41000/45000 Series adapters in Multi-Function mode
`sriov`	Enabled, Disabled
`number-of-vfs`	Integer value of modulo 8. Maximum values: `128` for a 3400/8400 Series adapter. For more information, see "Notes" on page 33. `240` for a 45000 Series adapter, with a maximum of `64` on one function. For more information, see "Notes" on page 33. `192` for a 41000 Series adapter, with a maximum of `64` on one function. For more information, see "Notes" on page 33.

Example

The following example shows the `adapter sriov set` command on a local host:

```
~# esxcli qlfastlinq qcc adapter sriov set -i Adapter_17_0
```

The following example shows the `adapter sriov set` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc adapter
sriov set -i Adapter_17_0
```

The output for both commands is:

```
SRIOV Configuration:
   Message: Success
   Mode: Single-Function
   Reboot Required: true
   SRIOV: Enabled
   Sriov Config:
         Pf Number: 0
         VFs Per PF: 24

         Pf Number: 1
         VFs Per PF: 32
```

The output shows the new SR-IOV setting, which becomes active after a reboot.

Adapter VPD

To view vital product data (VPD) for the selected adapter, issue the following command:

```
~# esxcli qlfastlinq qcc adapter vpd
```

Syntax

```
esxcli qlfastlinq qcc adapter vpd {cmd} [cmd options]
```

Available Commands

`get`
Gets VPD information for the specified Marvell Adapter. See Get VPD.

Get VPD

The `adapter vpd get` command displays VPD data read from the flash.

```
~# esxcli qlfastlinq qcc adapter vpd get
```

Syntax

```
esxcli qlfastlinq qcc adapter vpd get [cmd options]
```

Keywords

`-i|--identifier <str>` = Adapter identifier (required)

Parameter	Value (strings are case-sensitive)
`identifier`	Adapter identifier from the adapter `list` command

Example

The following example shows the `adapter vpd get` command on a local host:

```
~# esxcli qlfastlinq qcc adapter vpd get -i Adapter_33_0
```

The following example shows the `adapter vpd get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc adapter
vpd get -i Adapter_33_0
```

The output for both commands is:

```
VPD Configuration:
   Message: Success
   VPDData:
         Tag: EC
         Value: A-5618

         Tag: MN
         Value: 1590

         Tag: P#
         Value: HPE Eth 4x25Gb 1p 620QSFP28 Adapter

         Tag: PN
         Value: 817760-001

         Tag: ProductName
         Value: HPE Eth 4x25Gb 1p 620QSFP28 Adapter

         Tag: SN
         Value: 6789012345

         Tag: V0
         Value: 20W PCIeGen3x16

         Tag: V1
```

```
            Value: 4.1.3.6

            Tag: V2
            Value: 5901

.
.
.
            Tag: V7
            Value: 620QSFP28
```

Adapter Port Mode Configuration

Some 45000 Series QSFP-based adapters can be split from 1x100G to 4x25G or 4x10G, and from 1x40G or 2x40G to 4x10G speeds. On the 2x40G adapter, when set to 4x10G mode, the second 40G QSFP+ port is disabled.

The port mode command is used for this operation.

To set and display current port mode settings for these 45000 Series Adapters, issue the following command:

```
~# esxcli qlfastlinq qcc adapter portmode
```

Syntax

```
esxcli qlfastlinq qcc adapter portmode {cmd} [cmd options]
```

Available Commands

`get`
Shows port mode configuration for the specified Marvell Adapter. See Get Port Mode Configuration.

`set`
Saves port mode configuration for the specified Marvell Adapter. See Set Port-Mode Configuration.

Get Port Mode Configuration

To view port mode data read from the flash for a specific Marvell Adapter, issue the following command:

```
~# esxcli qlfastlinq qcc adapter portmode get
```

Syntax

```
esxcli qlfastlinq qcc adapter portmode get [cmd options]
```

Keywords

`-i|--identifier <str>` = Adapter identifier (required). Obtain the identifier from the adapter `list` command.

Example

The following example shows the `adapter portmode get` command on a local host:

```
~# esxcli qlfastlinq qcc adapter portmode get -i Adapter_4_0
```

The following example shows the `adapter portmode get` command on a remote host:

```
~# esxcli –s 172.27.9.144 –u root –p password qlfastlinq qcc
adapter portmode get -i Adapter_4_0
```

The output for both commands (on the 1x40G adapter) is:

```
Adapter Configuration:
   Message: Success
   Port Mode Data:
      Current Value: 1x40
      Description: Port Mode
      Name: PortMode
      Supported Values: 1x40,4x10
```

Set Port-Mode Configuration

To set port mode data in the flash for a specific Marvell Adapter, issue the following command:

```
~# esxcli qlfastlinq qcc adapter portmode set
```

Syntax

```
esxcli qlfastlinq qcc adapter portmode set [cmd options]
```

Keywords

`-i|--identifier <str>` = Adapter identifier (required). Obtain the identifier from the adapter `list` command output.

`-n|--name <str>` = Parameter name (required)

`-v|--value <str>` = Parameter value from list of supported values provided in the `get` command (required)

Example

The following example shows the `adapter portmode set` command on a local host:

```
~# esxcli qlfastlinq qcc adapter portmode set -i Adapter_4_0 – n
PortMode –v 4x10
```

The following example shows the `adapter portmode set` command on a remote host:

```
~# esxcli –s 172.27.9.144 –u root –p password qlfastlinq qcc
adapter portmode set -i Adapter_4_0 –n PortMode –v 4x10
```

The output for both commands (on the 1x40G adapter) is:

```
~# esxcli qlfastlinq qcc adapter portmode set -i Adapter_4_0 - n
PortMode –v 4x10
Adapter Configuration:
Message: Success
Port Mode Data:
     Current Value: 4x10
     Description: Port Mode
     Name: PortMode
     Supported Values: 1x40,4x10
Reboot Required: true
```

The output displays the new setting, which will be active after a reboot if the command is successful.

Adapter QinQ Configuration

QinQ, also referred to as *stacked VLANs* and *provider bridging,* is a VLAN feature that controls VLAN tagged transmit and receive frames based on various VLAN modes (Normal, Filtering, QinQ).

You can set these VLAN settings for each function (PF0–PF7, depending on the single function and NPAR modes) in the NVRAM.

These values will be used by drivers to pick/ignore transmit and receive VLAN tagged frames.

The QinQ commands set and display the VLAN QinQ table in the flash for 3400/8400 Series Adapters.

```
~# esxcli qlfastlinq qcc adapter qinq
```

Syntax

```
esxcli qlfastlinq qcc adapter qinq {cmd} [cmd options]
```

Available Commands

`get`
Shows QinQ configuration for the specified Marvell Adapter. See Get QinQ Configuration.

`setportcfg`
Sets QinQ port configuration in a file for the specified Marvell Adapter. See Set QinQ Port Configuration.

`setfunccfg`
Sets QinQ function configuration in a file for the specified Marvell Adapter. See Set QinQ Function Configuration

`set`
Saves QinQ configuration in the flash for the specified Marvell Adapter. See Set QinQ Configuration.

Get QinQ Configuration

To view QinQ VLAN table data read from the flash for a specific Marvell Adapter, issue the following command:

```
~# esxcli qlfastlinq qcc adapter qinq get
```

Syntax

```
esxcli qlfastlinq qcc adapter qinq get [cmd options]
```

Keywords

`-i|--identifier <str>` = Adapter identifier (required). Obtain the identifier from the adapter `list` command output.

Example

The following example shows the `adapter qinq get` command on a local host:

```
~# esxcli qlfastlinq qcc adapter qinq get -i Adapter_66_0
```

The following example shows the `adapter qinq get` command on a remote host:

```
~# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc
adapter qinq get -i Adapter_66_0
```

The output for both commands is:

```
QinQ Configuration:
Message: Success
Mode: Multi Function
QINQ:
```

```
Functions:
Function Number: 0
Vlan Id: 30
Vlan Id Pool: 1-64
Vlan Priority: 3

Function Number: 1
Vlan Id: 22
Vlan Id Pool: 65-128
Vlan Priority: 2

Function Number: 2
Vlan Id: 121
Vlan Id Pool: 129-192
Vlan Priority: 7

Function Number: 3
Vlan Id: 5
Vlan Id Pool: 193-256
Vlan Priority: 1

Function Number: 4
Vlan Id: 1
Vlan Id Pool: 257-320
Vlan Priority: 0

Function Number: 5
Vlan Id: 1
Vlan Id Pool: 321-384
Vlan Priority: 0

Function Number: 6
Vlan Id: 1
Vlan Id Pool: 385-448
Vlan Priority: 0

Function Number: 7
Vlan Id: 1
Vlan Id Pool: 449-510,600,701
Vlan Priority: 0
Ports:
Port Number: 0
```

```
Vlan Mode: Filtering

Port Number: 1
Vlan Mode: Filtering
```

Set QinQ Port Configuration

The `adapter qinq setportcfg` command saves user-provided port level QinQ VLAN table data in a temporary cache file for the specified Marvell Adapter. This cache file is used by the `set` command to save settings in the flash.

`~# esxcli qlfastlinq qcc adapter qinq setportcfg`

Syntax

`esxcli qlfastlinq qcc adapter qinq setportcfg [cmd options]`

Keywords

`-i|--identifier <str>` = Adapter identifier (required). Obtain the identifier from the adapter `list` command output.

`-p|--port-number <long>` = Port number (required). Physical port number of the adapter. Valid values are:

- `0` for Single port
- `0-1` for Dual port
- `0-3` for Quad port

`-m|--vlan-mode <str>` = VLAN mode (required). Valid values are:

- Normal
- Filtering
- QinQ

Example

The following example shows the `adapter qinq setportcfg` command on a local host:

`~# esxcli qlfastlinq qcc adapter qinq setportcfg -i Adapter_66_0 –p 0 –m QinQ`

The following example shows the `adapter qinq setportcfg` command on a remote host:

`~# esxcli –s 172.27.9.144 –u root –p password qlfastlinq qcc adapter qinq setportcfg -i Adapter_66_0 –p 0 –m QinQ`

The output for both commands is:

```
QinQ Configuration:
      Message: Success
```

```
Mode: Multi Function
QINQ:
   Functions:
   Function Number: 0
   Vlan Id: 30
   Vlan Id Pool: 1-64
   Vlan Priority: 3

   Function Number: 1
   Vlan Id: 22
   Vlan Id Pool: 65-128
   Vlan Priority: 2

   Function Number: 2
   Vlan Id: 121
   Vlan Id Pool: 129-192
   Vlan Priority: 7

   Function Number: 3
   Vlan Id: 5
   Vlan Id Pool: 193-256
   Vlan Priority: 1

   Function Number: 4
   Vlan Id: 1
   Vlan Id Pool: 257-320
   Vlan Priority: 0

   Function Number: 5
   Vlan Id: 1
   Vlan Id Pool: 321-384
   Vlan Priority: 0

   Function Number: 6
   Vlan Id: 10
   Vlan Id Pool: 385-448
   Vlan Priority: 0

   Function Number: 7
   Vlan Id: 1
   Vlan Id Pool: 449-510,600,701
   Vlan Priority: 0 Ports:
```

```
Port Number: 0 Vlan Mode: QinQ

Port Number: 1
Vlan Mode: Filtering
```

The command output is a combination of existing settings and user-modified settings saved in a cache file.

Set QinQ Function Configuration

The `adapter qinq setfunccfg` command saves user-provided, function-level QinQ VLAN table data in a temporary cache file for a specified Marvell Adapter. This cache file is used by the `set` command to save settings in the flash.

When issuing this command:

- All `setfunccfg` parameters (`vlan-id`, `vlan-priority`, `vlan-id-pool`) are read-only in normal VLAN mode and set to `0`.
- In QinQ mode:
  - `vlan-id-pool` is read only.
  - The `vlan-id` on all the functions of a port must be mutually exclusive.
  - The `vlan-id` must be nonzero.
- In Filtering mode:
  - There must be at least one valid `vlan-id` or `vlan-id-pool` value.
  - All parameters are read and write capable.
  - The maximum active VLAN IDs in the `vlan-id-pool` and `vlan-id` per adapter is 256.

The following command sets QinQ function configuration information for the specified Marvell Adapter in a file.

```
~# esxcli qlfastlinq qcc adapter qinq setfunccfg
```

Syntax

```
esxcli qlfastlinq qcc adapter qinq setfunccfg [cmd options]
```

Keywords

`-i|--identifier <str>` = Adapter identifier (required). Obtain the identifier from the adapter `list` command output.

`-n|--function-number <long>` = Function number (required). Function number of the adapter.

Valid values are in the range of `0-7`.

`-d|--vlan-id <long>` = VLAN ID

Valid values are in the range of `1–4094`.

`-l|--vlan-id-pool <str>` = VLAN ID pool. Numbers in string format. Numbers in a range are separated with a dash (`–`). Multiple numbers are separated with commas (`,`).

Valid values are in the range of `1–4094`.

`-p|--vlan-priority <long>` = VLAN priority.

Valid values are in the range of `0–7`.

Example

The following example shows the `adapter qinq setfunccfg` command on a local host:

```
~# esxcli qlfastlinq qcc adapter qinq setfunccfg -i Adapter_66_0
–n 7 –l 449-510,600,701
```

The following example shows the `adapter qinq setfunccfg` command on a remote host:

```
~# esxcli –s 172.27.9.144 –u root –p password qlfastlinq qcc
adapter qinq setfunccfg -i Adapter_66_0 –n 7 –l 449- 510,600,701
```

The output for both commands is:

```
QinQ Configuration:
      Message: Success
      Mode: Multi Function
      QINQ:
         Functions:
         Function Number: 0
         Vlan Id: 30
         Vlan Id Pool: 1-64
         Vlan Priority: 3

         Function Number: 1
         Vlan Id: 22
         Vlan Id Pool: 65-128
         Vlan Priority: 2

         Function Number: 2
         Vlan Id: 121
         Vlan Id Pool: 129-192
         Vlan Priority: 7

         Function Number: 3
```

```
Vlan Id: 5
Vlan Id Pool: 193-256
Vlan Priority: 1

Function Number: 4
Vlan Id: 1
Vlan Id Pool: 257-320
Vlan Priority: 0

Function Number: 5
Vlan Id: 1
Vlan Id Pool: 321-384
Vlan Priority: 0

Function Number: 6
Vlan Id: 10
Vlan Id Pool: 385-448
Vlan Priority: 0

Function Number: 7
Vlan Id: 1
Vlan Id Pool: 449-510,600,701
Vlan Priority: 0 Ports:
Port Number: 0 Vlan Mode: QinQ

Port Number: 1
Vlan Mode: Filtering
```

The command output is a combination of existing settings and user-modified settings saved in a cache file.

Set QinQ Configuration

The `adapter qinq set` command imports user-provided data from a temporary cache file to the flash. This command fails if cache file is not present. When the command completes successfully, the cache file is deleted.

To import QinQ VLAN table data for a specified Marvell Adapter, issue the following command:

```
~# esxcli qlfastlinq qcc adapter qinq set
```

Syntax

```
esxcli qlfastlinq qcc adapter qinq set [cmd options]
```

Keywords

`-i|--identifier <str>` = Adapter identifier (required). Obtain the identifier from the adapter `list` command output.

Example

The following example shows the `adapter qinq set` command on a local host:

```
~# esxcli qlfastlinq qcc adapter qinq set -i Adapter_66_0
```

The following example shows the `adapter qinq set` command on a remote host:

```
~# esxcli –s 172.27.9.144 –u root –p password qlfastlinq qcc
adapter qinq set -i Adapter_66_0
```

The output for both commands is:

```
QinQ Configuration:
      Message: Success
      Mode: Multi Function
      QINQ:
         Functions:
         Function Number: 0
         Vlan Id: 30
         Vlan Id Pool: 1-64
         Vlan Priority: 3

         Function Number: 1
         Vlan Id: 22
         Vlan Id Pool: 65-128
         Vlan Priority: 2

         Function Number: 2
         Vlan Id: 121
         Vlan Id Pool: 129-192
         Vlan Priority: 7

         Function Number: 3
         Vlan Id: 5
         Vlan Id Pool: 193-256
         Vlan Priority: 1

         Function Number: 4
         Vlan Id: 1
```

```
            Vlan Id Pool: 257-320
            Vlan Priority: 0

            Function Number: 5
            Vlan Id: 1
            Vlan Id Pool: 321-384
            Vlan Priority: 0

            Function Number: 6
            Vlan Id: 1
            Vlan Id Pool: 385-448
            Vlan Priority: 0

            Function Number: 7
            Vlan Id: 1
            Vlan Id Pool: 449-510,600,701
            Vlan Priority: 0 Ports:
            Port Number: 0
            Vlan Mode: Filtering

            Port Number: 1
            Vlan Mode: Filtering
Reboot Required: true
```

The output shows the new settings, which will be active after a reboot if the command is successful.

Adapter Port Namespace Commands and Sub-namespaces

This section provides QESXCLI information on the following:

- Adapter Port List
- Adapter Port Information
- Adapter Port FCoE Boot Configuration (3400/8400 Series)
- Adapter Port FCoE Boot Configuration (41000/45000 Series)
- Adapter Port iSCSI Boot Configuration (3400/8400 Series)
- Adapter Port iSCSI Boot Configuration (41000/45000 Series)
- Adapter Port MBA Boot Configuration Namespace Commands and Sub-namespaces
- Adapter Port Diagnostics Namespace Commands and Sub-namespaces

- Adapter Port DCBX Namespace Commands and Sub-namespaces
- Adapter Port Link Settings (SmartAN) Namespace Commands and Sub-namespaces

```
~# esxcli qlfastlinq qcc port
```

Syntax

```
esxcli qlfastlinq qcc port {cmd} [cmd options]
```

Namespaces

Available namespaces include the following:

Namespace	Description
`dcbx`	QCC CLI for port-level DCBX information
`diag`	QCC CLI for port-level diagnostics
`dmi`	QCC CLI for port-level SFP DMI information
`fcoebootcfg579xx`	QCC CLI for port FCoE boot configuration information for 41000/45000 Series adapters
`fcoebootcfg`	QCC CLI for port FCoE boot configuration information for 3400/8400 Series adapters
`info`	QCC CLI for port information
`iscsibootcfg`	QCC CLI for port iSCSI boot configuration information for 3400/8400 Series adapters
`iscsibootcfg579xx`	QCC CLI for port iSCSI boot configuration information for 41000/45000 Series Adapters
`linkSettings`	QCC CLI for port-level Advanced Link Settings information
`mbabootcfg`	QCC CLI for port MBA boot configuration information

Available Commands

`list`
Shows identifiers for Marvell Adapter ports discovered on the host.

Adapter Port List

The `port list` command (shown issued from a local host) lists all of the supported 3400/8400 Series and 41000/45000 Series ports, as follows:

```
~# esxcli qlfastlinq qcc port list
```

The following example shows the `port list` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
list
```

The output for both commands is:

```
Ports
-----------
Port_10_0_0
Port_10_0_1
Port_16_0_0
Port_16_0_1
Port_27_0_0
Port_27_0_1
```

Adapter Port Information

The `port info` command shows information about the selected Marvell Adapter port:

```
~# esxcli qlfastlinq qcc port info
```

Syntax

```
esxcli qlfastlinq qcc port info {cmd} [cmd options]
```

Available Commands

`get`
Shows information about the specified Marvell Adapter port. See Get Port Information.

Get Port Information

The `port info get` command shows port-level information for the specified Marvell Adapter port.

```
~# esxcli qlfastlinq qcc port info get
```

Syntax

```
esxcli qlfastlinq qcc port info get [cmd options]
```

Keywords

`-i|--identifier <str>` = Adapter identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port info get` command on a local host:

```
~# esxcli qlfastlinq qcc port info get -i Port_10_0_0
```

The following example shows the `port info get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
info get -i Port_10_0_0
```

The output for both commands is:

```
Port Information:
   Message: Success
Port Info:
      Bus Number: 18
      Description: BCM57810 10 Gigabit Ethernet Multi Function
      Device Number: 0
      Duplex: Half
      Fcoe Driver Version: NA
      Fec Auto Neg Mode: NA
      Fec Force Mode: NA
      Iscsi Driver Version: NA
      Link Settings: NA
      Link Settings Link Speed: NA
      Link Speedin Mbps: Unknown
      Link State: Link Down
      Media Type: Optical
      Nic Driver Version: 1.0.69.0
      Port Number: 0
      Title: BCM57810 10 Gigabit Ethernet Multi Function rev 10 (vmnic20)
```

Adapter Port FCoE Boot Configuration (3400/8400 Series)

The `fcoebootcfg` command performs FCoE boot configuration operations on 3400/8400 Series Adapters. For FCoE boot configuration on 41000/45000 Series Adapters, see Adapter Port FCoE Boot Configuration (41000/45000 Series).

```
~# esxcli qlfastlinq qcc port fcoebootcfg
```

Syntax

```
esxcli qlfastlinq qcc port fcoebootcfg {cmd} [cmd options]
```

Available Commands

`get`
Shows FCoE boot configuration information for the specified Marvell Adapter (3400/8400 Series Adapter) port. See Get FCoE Boot Configuration.

`setgenparams`
Saves FCoE boot configuration general parameters for the specified Marvell Adapter (3400/8400 Series adapter) port in a file. See Set FCoE Boot Configuration General Parameters.

`settgtparams`
Saves FCoE boot configuration target parameters for the specified Marvell Adapter (3400/8400 Series adapter) port in a file. See Set FCoE Boot Configuration Target Parameters.

`set`
Sets FCoE boot configuration information for the specified Marvell Adapter (3400/8400 Series adapter) port. See Set FCoE Boot Configuration.

Get FCoE Boot Configuration

To view FCoE boot configuration data for the specified Marvell Adapter (3400/8400 Series Adapter) port, issue the `port fcoebootcfg get` command as follows:

```
~# esxcli qlfastlinq qcc port fcoebootcfg get
```

Syntax

```
esxcli qlfastlinq qcc port fcoebootcfg get [cmd options]
```

Keywords

`-i|--identifier <str>` = Adapter identifier (required). Obtain the identifier from the port `list` output

Example

The following example shows the `port fcoebootcfg get` command on a local host:

```
~# esxcli qlfastlinq qcc port fcoebootcfg get -i Port_27_0_0
```

The following example shows the `port fcoebootcfg get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
fcoebootcfg get -i Port_27_0_0
```

The output for both commands is:

```
FCoE Boot Configuration:
   FCOEBoot Info:
         Boot Targets:
         Boot LUN: 1
         Connect: Enabled
```

```
            Port WWN: 50-01-00-0E-0F-0A-0B-0C
            Target Number: 0

            Boot LUN: 3
            Connect: Enabled
            Port WWN: 20-01-00-0E-0A-04-0B-06
            Target Number: 1
.
.
.
            Boot LUN: 0
            Connect: Disabled
            Port WWN: 00-00-00-00-00-00-00-00
            Target Number: 6

            Boot LUN: 0
            Connect: Disabled
            Port WWN: 00-00-00-00-00-00-00-00
            Target Number: 7
            General Parameters:
            Boot To Fcoe Target: One Time Disabled
            Fabric Discovery Timeout: 5
            Fcoe Hba Boot Mode: Disabled
            Link Up Delay Time: 10
            Lun Busy Retry Count: 9
            Target As First HDD: Disabled
   Message: Success
```

Set FCoE Boot Configuration General Parameters

The `port fcoebootcfg setgenparams` command saves user-provided FCoE general parameter data for the specified Marvell Adapter (3400/8400 Series adapter) port in a temporary cache file. The cache file is used by the `set` command to save settings in the flash.

```
~# esxcli qlfastlinq qcc port fcoebootcfg setgenparams
```

Syntax

```
esxcli qlfastlinq qcc port fcoebootcfg setgenparams [cmd options]
```

Keywords

```
     -o|--boot-to-fcoe-target <str> = Boot to FCoE target
-d|--fabric-discovery-timeout <long> = Fabric discovery timeout
       -m|--fcoe-hba-boot-mode <str> = FCoE HBA boot mode
               -i|--identifier <str> = Port identifier (required)
     -u|--link-up-delay-time <long> = Link up delay time
   -r|--lun-busy-retry-count <long> = LUN busy retry count
     -f|--target-as-first-hdd <str> = Target as first HDD
```

Parameter	Value (strings are case-sensitive)
`identifier`	Port identifier from the port `list` command
`boot-to-fcoe-target`	Enabled, Disabled, OneTimeDisabled
`fcoe-hba-boot-mode`	Enabled, Disabled
`target-as-first-hdd`	Enabled, Disabled
`fabric-discovery-timeout`	Integer value (`1–8`)
`link-up-delay-time`	Integer value (`0–255`)
`lun-busy-retry-count`	Integer value (`0–60`)

Example

The following example shows the `port fcoebootcfg setgenparams` command on a local host:

```
~# esxcli qlfastlinq qcc port fcoebootcfg setgenparams -i
Port_27_0_0 -r 20
```

The following example shows the `port fcoebootcfg setgenparams` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
fcoebootcfg setgenparams -i Port_27_0_0 -r 20
```

The output for both commands is:

```
FCoE Boot Configuration:
   FCOEBoot Info:
         Boot Targets:
         Boot LUN: 1
         Connect: Enabled
         Port WWN: 50-01-00-0E-0F-0A-0B-0C
         Target Number: 0
```

```
            Boot LUN: 3
            Connect: Enabled
            Port WWN: 20-01-00-0E-0A-04-0B-06
            Target Number: 1
.
.
.
            Boot LUN: 0
            Connect: Disabled
            Port WWN: 00-00-00-00-00-00-00-00
            Target Number: 6

            Boot LUN: 0
            Connect: Disabled
            Port WWN: 00-00-00-00-00-00-00-00
            Target Number: 7
            General Parameters:
            Boot To Fcoe Target: One Time Disabled
            Fabric Discovery Timeout: 5
            Fcoe Hba Boot Mode: Disabled
            Link Up Delay Time: 10
            Lun Busy Retry Count: 20
            Target As First HDD: Disabled
   Message: Success
```

The output displayed is combination of existing settings and user modified settings saved in the cache file. Same data is present in cache file.

Set FCoE Boot Configuration Target Parameters

The `port fcoebootcfg settgtparams` command saves user-provided FCoE target parameters data for a specified Marvell Adapter port in a temporary cache file. This cache file will be used by the `set` command to save the settings in the flash.

```
~# esxcli qlfastlinq qcc port fcoebootcfg settgtparams
```

Syntax

```
esxcli qlfastlinq qcc port fcoebootcfg settgtparams [cmd options]
```

Keywords

`-l|--boot-lun <long>` = Link up delay time
`-c|--connect <str>` = Connect
`-i|--identifier <str>` = Port identifier (required)
`-p|--port-wwn <str>` = Hyphen-separated port WWN (for example, `20-01-0E-0A-04-06-0E-0B`)
`-n|--target-number <long>` = Target number (required)

Parameter	Value (strings are case-sensitive)
`identifier`	Port identifier from the port `list` command
`target-number`	Target number whose parameter to be set (`0-7`)
`connect`	Enabled, Disabled
`port-wwn`	Hyphen-separated port WWN (for example, `20-01-0E-0A-04-06-0E-0B`)
`boot-lun`	Boot LUN (`0-65535`)

Example

The following example shows the `port fcoebootcfg settgtparams` command on a local host:

```
~# esxcli qlfastlinq qcc port fcoebootcfg settgtparams -i
Port_27_0_0 -n 1 -l 4
```

The following example shows the `port fcoebootcfg settgtparams` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
fcoebootcfg settgtparams -i Port_27_0_0 -n 1 -l 4
```

The output for both commands is:

```
FCoE Boot Configuration:
   FCOEBoot Info:
         Boot Targets:
         Boot LUN: 1
         Connect: Enabled
         Port WWN: 50-01-00-0E-0F-0A-0B-0C
         Target Number: 0

         Boot LUN: 4
         Connect: Enabled
         Port WWN: 20-01-00-0E-0A-04-0B-06
```

```
            Target Number: 1
.
.
.
            Boot LUN: 0
            Connect: Disabled
            Port WWN: 00-00-00-00-00-00-00-00
            Target Number: 7
            General Parameters:
            Boot To Fcoe Target: One Time Disabled
            Fabric Discovery Timeout: 5
            Fcoe Hba Boot Mode: Disabled
            Link Up Delay Time: 10
            Lun Busy Retry Count: 20
            Target As First HDD: Disabled
   Message: Success
```

The output shown is combination of existing settings and user-modified settings saved in the cache file. The same data is present in the cache file.

Set FCoE Boot Configuration

To import user-provided FCoE boot configuration data for the specified Marvell Adapter (3400/8400 Series Adapter) from a temporary cache file to the flash, issue the `fcoebootcfg set` command as follows:

```
~# esxcli qlfastlinq qcc port fcoebootcfg set
```

This command fails if the cache file is not present. On successful completion of the `set` command, the cache file is deleted.

Syntax

```
esxcli qlfastlinq qcc port fcoebootcfg set [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output

Example

The following example shows the `port fcoebootcfg set` command on a local host:

```
~# esxcli qlfastlinq qcc port fcoebootcfg set -i Port_27_0_0
```

The following example shows the `port fcoebootcfg set` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
fcoebootcfg set -i Port_27_0_0
```

The output for both commands is:

```
FCoE Boot Configuration:
   FCOEBoot Info:
         Boot Targets:
         Boot LUN: 1
         Connect: Enabled
         Port WWN: 50-01-00-0E-0F-0A-0B-0C
         Target Number: 0

         Boot LUN: 4
         Connect: Enabled
         Port WWN: 20-01-00-0E-0A-04-0B-06
         Target Number: 1
.
.
.
         Boot LUN: 0
         Connect: Disabled
         Port WWN: 00-00-00-00-00-00-00-00
         Target Number: 7
         General Parameters:
         Boot To Fcoe Target: One Time Disabled
         Fabric Discovery Timeout: 5
         Fcoe Hba Boot Mode: Disabled
         Link Up Delay Time: 10
         Lun Busy Retry Count: 20
         Target As First HDD: Disabled
   Message: Success
   Reboot Required: true
```

The output displayed is new setting, which will be active after reboot if the command is successful.

Adapter Port FCoE Boot Configuration (41000/45000 Series)

The `port fcoebootcfg579xx` command performs FCoE boot configuration operations on 41000/45000 Series Adapters. For FCoE boot configuration on 3400/8400 Series Adapters, see Adapter Port FCoE Boot Configuration (3400/8400 Series).

```
~# esxcli qlfastlinq qcc port fcoebootcfg579xx
```

Syntax

```
esxcli qlfastlinq qcc port fcoebootcfg579xx {cmd} [cmd options]
```

Available Commands

`get`
Shows FCoE boot configuration information for the specified Marvell Adapter (41000/45000 Series) port. See Get FCoE Boot Configuration.

`setgenparams`
Saves FCoE boot configuration general parameters for the specified Marvell Adapter (41000/45000 Series) port in a file. See Set FCoE Boot Configuration General Parameters.

`settgtparams`
Saves FCoE boot configuration target parameters for the specified Marvell Adapter (41000/45000 Series) port in a file. See Set FCoE Boot Configuration Target Parameters.

`set`
Imports FCoE boot configuration information for the specified Marvell Adapter (41000/45000 Series) port. See Set FCoE Boot Configuration.

Get FCoE Boot Configuration

To view FCoE boot configuration data for the specified Marvell Adapter (41000/45000 Series) port, issue the `port fcoebootcfg579xx get` command as follows:

```
~# esxcli qlfastlinq qcc port fcoebootcfg579xx get
```

Syntax

```
esxcli qlfastlinq qcc port fcoebootcfg579xx get [cmd options]
```

Keywords

`-i|--identifier <str>` = Adapter identifier (required). Obtain the identifier from the adapter `port list` output

Example

The following example shows the `port fcoebootcfg579xx get` command on a local host:

```
~# esxcli qlfastlinq qcc port fcoebootcfg579xx get -i Port_17_0_0
```

The following example shows the `port fcoebootcfg579xx get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
fcoebootcfg579xx get -i Port_17_0_0
```

The output for both commands is:

```
FCoE Boot Configuration:
   FCOEBoot Info:
         Boot Targets:
         Boot LUN: 0
         Connect: Disabled
         Port WWN: 00-00-00-00-00-00-00-00
         Target Number: 0

         Boot LUN: 0
         Connect: Disabled
         Port WWN: 00-00-00-00-00-00-00-00
         Target Number: 1
.
.
.
         Boot LUN: 0
         Connect: Disabled
         Port WWN: 00-00-00-00-00-00-00-00
         Target Number: 7
         General Parameters:
         Boot Protocol: None
         Fabric Login Retry Count: 5
         Fip Vlan: 0
         Target Login Retry Count: 5
   Message: Success
```

Set FCoE Boot Configuration General Parameters

The `port fcoebootcfg579xx setgenparams` command saves user-provided FCoE general parameter data for the specified Marvell Adapter (41000/45000 Series Adapter) port in a temporary cache file. The cache file is used by the `set` command to save settings in the flash.

```
~# esxcli qlfastlinq qcc port fcoebootcfg579xx setgenparams
```

Syntax

```
esxcli qlfastlinq qcc port fcoebootcfg579xx setgenparams [cmd
options]
```

Keywords

```
            -o|--boot-protocol <str> = Boot protocol
-f|--fabric-login-retry-count <long> = Fabric login retry count
                -v|--fip-vlan <long> = FIP VLAN
              -i|--identifier <str> = Port identifier (required)
-r|--target-login-retry-count <long> = Target login retry count
```

Parameter	Value (strings are case-sensitive)
`identifier`	Port identifier from the port `list` command
`boot-protocol`	FCOE, None
`fabric-login-retry-count`	Integer value (`0–7`)
`target-login-retry-count`	Integer value (`0–255`)
`fip-vlan`	Integer value (`0–4094`)

Example

The following example shows the `port fcoebootcfg579xx setgenparams` command on a local host:

```
~# esxcli qlfastlinq qcc port fcoebootcfg579xx setgenparams -i
Port_17_0_0 -v 4
```

The following example shows the `port fcoebootcfg579xx setgenparams` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
fcoebootcfg579xx setgenparams -i Port_17_0_0 -v 4
```

The output for both commands is:

```
FCoE Boot Configuration:
   FCOEBoot Info:
```

```
            Boot Targets:
            Boot LUN: 0
            Connect: Disabled
            Port WWN: 00-00-00-00-00-00-00-00
            Target Number: 0

            Boot LUN: 0
            Connect: Disabled
            Port WWN: 00-00-00-00-00-00-00-00
            Target Number: 1

            Boot LUN: 0
            Connect: Disabled
            Port WWN: 00-00-00-00-00-00-00-00
            Target Number: 7
            General Parameters:
            Boot Protocol: None
            Fabric Login Retry Count: 5
            Fip Vlan: 4
            Target Login Retry Count: 5
   Message: Success
```

The command output includes a combination of existing settings and user-modified settings saved in the cache file.

Set FCoE Boot Configuration Target Parameters

The `port fcoebootcfg579xx settgtparams` command sets user-provided FCoE target parameters data for a specified Marvell Adapter port in a temporary cache file. This cache file will be used by the `set` command to save the settings in the flash.

```
~# esxcli qlfastlinq qcc port fcoebootcfg579xx settgtparams
```

Syntax

```
esxcli qlfastlinq qcc port fcoebootcfg579xx settgtparams [cmd
options]
```

Keywords

`-l|--boot-lun <long>` = Boot LUN

`-c|--connect <str>` = Connect

`-i|--identifier <str>` = Port identifier (required)

`-p|--port-wwn <str>` = Hyphen-separated port WWN (for example, `20-01-0E-0A-04-06-0E-0B`)
`-n|--target-number <long>` = Target number (required)

Parameter	Value (strings are case-sensitive)
`identifier`	Port identifier from the port `list` command
`target-number`	Target number whose parameter to be set (`0-7`)
`connect`	Enabled, Disabled
`port-wwn`	Hyphen-separated port WWN (for example, `20-01-0E-0A-04-06-0E-0B`)
`boot-lun`	Boot LUN (`0-65535`)

Example

The following example shows the `port fcoebootcfg579xx settgtparams` command on a local host:

```
~# esxcli qlfastlinq qcc port fcoebootcfg579xx settgtparams -i
Port_17_0_0 -n 0 -c Enabled
```

The following example shows the `port fcoebootcfg579xx settgtparams` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
fcoebootcfg579xx settgtparams -i Port_17_0_0 -n 0 -c Enabled
```

The output for both commands is:

```
FCoE Boot Configuration:
   FCOEBoot Info:
         Boot Targets:
         Boot LUN: 0
         Connect: Enabled
         Port WWN: 00-00-00-00-00-00-00-00
         Target Number: 0

         Boot LUN: 0
         Connect: Disabled
         Port WWN: 00-00-00-00-00-00-00-00
         Target Number: 1
.
.
.
```

```
            Boot LUN: 0
            Connect: Disabled
            Port WWN: 00-00-00-00-00-00-00-00
            Target Number: 7
            General Parameters:
            Boot Protocol: None
            Fabric Login Retry Count: 5
            Fip Vlan: 4
            Target Login Retry Count: 5
   Message: Success
```

The output shows a combination of existing settings and user-modified settings saved in the cache file.

Set FCoE Boot Configuration

To import user-provided FCoE boot configuration data for the specified Marvell Adapter (41000/45000 Series Adapter) from a temporary cache file to the flash, issue the `port fcoebootcfg set` command as follows:

```
~# esxcli qlfastlinq qcc port fcoebootcfg579xx set
```

This command fails if the cache file is not present. On successful completion of the `set` command, the cache file is deleted.

Syntax

```
esxcli qlfastlinq qcc port fcoebootcfg579xx set [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output

Example

The following example shows the `port fcoebootcfg579xx set` command on a local host:

```
~# esxcli qlfastlinq qcc port fcoebootcfg579xx set -i Port_17_0_0
```

The following example shows the `port fcoebootcfg579xx set` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
fcoebootcfg579xx set -i Port_17_0_0
```

The output for both commands is:

```
FCoE Boot Configuration:
   FCOEBoot Info:
```

```
            Boot Targets:
            Boot LUN: 0
            Connect: Enabled
            Port WWN: 00-00-00-00-00-00-00-00
            Target Number: 0

            Boot LUN: 0
            Connect: Disabled
            Port WWN: 00-00-00-00-00-00-00-00
            Target Number: 1
.
.
.
            Boot LUN: 0
            Connect: Disabled
            Port WWN: 00-00-00-00-00-00-00-00
            Target Number: 7
            General Parameters:
            Boot Protocol: None
            Fabric Login Retry Count: 5
            Fip Vlan: 4
            Target Login Retry Count: 5
   Message: Success
   Reboot Required: true
```

The output displayed is new setting, which will be active after reboot if command is successful.

Adapter Port iSCSI Boot Configuration (3400/8400 Series)

The following command performs iSCSI boot configuration operations on 3400/8400 Series adapters:

```
~# esxcli qlfastlinq qcc port iscsibootcfg
```

Syntax

```
esxcli qlfastlinq qcc port iscsibootcfg {cmd} [cmd options]
```

Available Commands

`get`
Shows iSCSI boot configuration information for a specified Marvell Adapter (3400/8400 Series) port. See Get iSCSI Boot Configuration.

`setgenparams`
Saves iSCSI boot configuration general parameters for a specified Marvell Adapter (3400/8400 Series) port in a file. See Set iSCSI Boot Configuration General Parameters.

`setinitparams`
Saves iSCSI boot configuration initiator parameters for a specified Marvell Adapter (3400/8400 Series port in a file. See Set iSCSI Boot Configuration Initiator Parameters.

`settgtparams`
Saves iSCSI boot configuration target parameters for a specified Marvell Adapter (3400/8400 Series) port in a file. See Set iSCSI Boot Configuration Target Parameters.

`setmpioparams`
Saves iSCSI boot configuration MPIO parameters for a specified Marvell Adapter (3400/8400 Series) port in a file. See Set iSCSI Boot Configuration MPIO Parameters.

`set`
Imports iSCSI boot configuration information for a specified Marvell Adapter (3400/8400 Series) port. See Set iSCSI Boot Configuration.

Get iSCSI Boot Configuration

To show iSCSI boot configuration information for a specified Marvell Adapter (3400/8400 Series) port, issue the following command:

```
~# esxcli qlfastlinq qcc port iscsibootcfg get
```

Syntax

```
esxcli qlfastlinq qcc port iscsibootcfg get [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port iscsibootcfg get` command on a local host:

```
~# esxcli qlfastlinq qcc port iscsibootcfg get -i Port_27_0_0
```

The following example shows the `port iscsibootcfg get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
iscsibootcfg get -i Port_27_0_0
```

The output for both commands is:

```
iSCSI Boot Configuration:
   ISCSIBoot Info:
         Boot Targets:
         Connect: Enabled
         Target Boot LUN: 5
         Target Chap Id:
         Target Chap Secret:
         Target IPAddress: 192.168.100.227
         Target Iscsi Name:
         Target Number: 0
         Target Tcp Port: 3260

         Connect: Disabled
         Target Boot LUN: 0
         Target Chap Id:
         Target Chap Secret:
         Target IPAddress: 0.0.0.0
         Target Iscsi Name:
         Target Number: 1
         Target Tcp Port: 0
         General Parameters:
         Boot To Iscsi Target: Enabled
         Chap Authentication: Disabled
         Dhcp Vendor Id: BRCM
         Hba Boot Mode: Disabled
         Ip Version: IPv4
         Iscsi Parameters Via DHCP: Disabled
         Link Up Delay Time: 0
         Lun Busy Retry Count: 0
         Target As First HDD: Disabled
         Tcp Ip Parameters Via DHCP: Enabled
         Use Tcp Timestamp: Disabled
         Initiator Parameters:
         Default Gateway: 0.0.0.0
         Initiator Chap Id:
         Initiator Chap Secret:
         Initiator IPAddress: 192.168.100.225
         Initiator Iscsi Name: iqn.1995-05.com.broadcom.iscsiboot
         Primary DNS: 0.0.0.0
```

```
            Secondary DNS: 0.0.0.0
            Subnet Mask: 0.0.0.0
            Mpio Parameters:
            MACAddress: 00:00:00:00:00:00
            Mpio Mode: Enabled
            Use Independent Target Name: Disabled
            Use Independent Target Portal: Enabled
      Message: Success
```

Set iSCSI Boot Configuration General Parameters

To save user-provided iSCSI boot configuration general parameters information for a specified Marvell Adapter (3400/8400 Series) port to a temporary cache file, issue the following command:

```
~# esxcli qlfastlinq qcc port iscsibootcfg setgenparams
```

Syntax

```
esxcli qlfastlinq qcc port iscsibootcfg setgenparams [cmd options]
```

Keywords

```
-o|--boot-to-iscsi-target <str> = Boot to iSCSI target
 -a|--chap-authentication <str> = CHAP authentication
      -d|--dhcp-vendor-id <str> = DHCP vendor ID
       -m|--hba-boot-mode <str> = HBA boot mode
          -i|--identifier <str> = Port identifier (required)
          -v|--ip-version <str> = IP version
      -s|--iscsi-via-dhcp <str> = iSCSI parameters via DHCP
  -u|--link-up-delay-time <long> = Link up delay time
-r|--lun-busy-retry-count <long> = LUN busy retry count
  -f|--target-as-first-hdd <str> = Target as first HDD
      -c|--tcpip-via-dhcp <str> = TCP/IP parameters via DHCP
    -e|--use-tcp-timestamp <str> = Use TCP timestamp
```

Parameter	Value (strings are case-sensitive)
`identifier`	Port identifier from the port `list` command
`boot-to-iscsi-target`	Target number whose parameter to be set (`0-7`)
`connect`	Enabled, Disabled, OneTimeDisabled

Parameter	Value (strings are case-sensitive)
chap-authentication	Enabled, Disabled
hba-boot-mode	Enabled, Disabled
iscsi-via-dhcp	Enabled, Disabled
target-as-first-hdd	Enabled, Disabled
tcpip-via-dhcp	Enabled, Disabled
use-tcp-timestamp	Enabled, Disabled
dhcp-vendor-id	String value
ip-version	IPv4, IPv6
link-up-delay-time	Integer value (0–255)
lun-busy-retry-count	Integer value (0–60)

Example

The following example shows the `port iscsibootcfg setgenparams` command on a local host:

```
~# esxcli qlfastlinq qcc port iscsibootcfg setgenparams -i
Port_27_0_0 -u 55
```

The following example shows the `port iscsibootcfg setgenparams` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
iscsibootcfg setgenparams -i Port_27_0_0 -u 55
```

The output for both commands is:

```
iSCSI Boot Configuration:
   ISCSIBoot Info:
         Boot Targets:
         Connect: Enabled
         Target Boot LUN: 5
         Target Chap Id:
         Target Chap Secret:
         Target IPAddress: 192.168.100.227
         Target Iscsi Name:
         Target Number: 0
         Target Tcp Port: 3260

         Connect: Disabled
         Target Boot LUN: 0
         Target Chap Id:
```

```
                Target Chap Secret:
                Target IPAddress: 0.0.0.0
                Target Iscsi Name:
                Target Number: 1
                Target Tcp Port: 0
                General Parameters:
                Boot To Iscsi Target: Enabled
                Chap Authentication: Disabled
                Dhcp Vendor Id: BRCM
                Hba Boot Mode: Disabled
                Ip Version: IPv4
                Iscsi Parameters Via DHCP: Disabled
                Link Up Delay Time: 55
                Lun Busy Retry Count: 0
                Target As First HDD: Disabled
                Tcp Ip Parameters Via DHCP: Enabled
                Use Tcp Timestamp: Disabled
                Initiator Parameters:
                Default Gateway: 0.0.0.0
                Initiator Chap Id:
                Initiator Chap Secret:
                Initiator IPAddress: 192.168.100.225
                Initiator Iscsi Name: iqn.1995-05.com.broadcom.iscsiboot
                Primary DNS: 0.0.0.0
                Secondary DNS: 0.0.0.0
                Subnet Mask: 0.0.0.0
                Mpio Parameters:
                MACAddress: 00:00:00:00:00:00
                Mpio Mode: Enabled
                Use Independent Target Name: Disabled
                Use Independent Target Portal: Enabled
   Message: Success
```

The command output contains a combination of existing settings and the user-modified settings saved in the cache file.

Set iSCSI Boot Configuration Initiator Parameters

To save user-provided iSCSI boot configuration initiator parameters information for a specified Marvell Adapter (3400/8400 Series) port to a temporary cache file, issue the following command:

```
~# esxcli qlfastlinq qcc port iscsibootcfg setinitparams
```

Syntax

```
esxcli qlfastlinq qcc port iscsibootcfg setinitparams [cmd
options]
```

Keywords

```
        -c|--chap-id <str> = Initiator CHAP ID
    -w|--chap-secret <str> = Initiator CHAP secret
-g|--default-gateway <str> = Initiator gateway address [a]
     -i|--identifier <str> = Port identifier (required)
     -a|--ip-address <str> = Initiator IP address [a]
     -q|--iscsi-name <str> = Initiator iSCSI name
    -p|--primary-dns <str> = Primary DNS address [a]
  -d|--secondary-dns <str> = Secondary DNS address [a]
    -m|--subnet-mask <str> = Initiator subnet mask address [a]
```

[a] Addresses can contain either periods (`.`) or colons (`:`) to separate segments.

Parameter	Value (strings are case-sensitive)
`identifier`	Port identifier from the port `list` command
`chap-id`	CHAP ID string
`chap-secret`	CHAP password string
`ip-address`	Initiator IP address [a]
`subnet-mask`	Initiator subnet mask [a]
`default-gateway`	Initiator default gateway [a]
`iscsi-name`	Initiator iSCSI qualified name (IQN)
`primary-dns`	Primary DNS address [a]
`secondary-dns`	Secondary DNS address [a]

[a] Addresses can contain either periods (`.`) or colons (`:`) to separate segments.

Example

The following example shows the `port iscsibootcfg setinitparams` command on a local host:

```
~# esxcli qlfastlinq qcc port iscsibootcfg setinitparams -i
Port_27_0_0 -a 192.168.100.225
```

The following example shows the `port iscsibootcfg setinitparams` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
iscsibootcfg setinitparams -i Port_27_0_0 -a 192.168.100.225
```

The output for both commands is:

```
iSCSI Boot Configuration:
   ISCSIBoot Info:
         Boot Targets:
         Connect: Enabled
         Target Boot LUN: 5
         Target Chap Id:
         Target Chap Secret:
         Target IPAddress: 192.168.100.227
         Target Iscsi Name:
         Target Number: 0
         Target Tcp Port: 3260

         Connect: Disabled
         Target Boot LUN: 0
         Target Chap Id:
         Target Chap Secret:
         Target IPAddress: 0.0.0.0
         Target Iscsi Name:
         Target Number: 1
         Target Tcp Port: 0
         General Parameters:
         Boot To Iscsi Target: Enabled
         Chap Authentication: Disabled
         Dhcp Vendor Id: BRCM
         Hba Boot Mode: Disabled
         Ip Version: IPv4
         Iscsi Parameters Via DHCP: Disabled
         Link Up Delay Time: 55
         Lun Busy Retry Count: 0
         Target As First HDD: Disabled
         Tcp Ip Parameters Via DHCP: Enabled
         Use Tcp Timestamp: Disabled
         Initiator Parameters:
         Default Gateway: 0.0.0.0
```

```
            Initiator Chap Id:
            Initiator Chap Secret:
            Initiator IPAddress: 192.168.100.225
            Initiator Iscsi Name: iqn.1995-05.com.broadcom.iscsiboot
            Primary DNS: 0.0.0.0
            Secondary DNS: 0.0.0.0
            Subnet Mask: 0.0.0.0
            Mpio Parameters:
            MACAddress: 00:00:00:00:00:00
            Mpio Mode: Enabled
            Use Independent Target Name: Disabled
            Use Independent Target Portal: Enabled
   Message: Success
```

The command output is a combination of existing settings and user-modified settings saved in the cache file.

Set iSCSI Boot Configuration Target Parameters

To save user-provided iSCSI boot configuration target parameters information for a specified Marvell Adapter (3400/8400 Series) port to a temporary cache file, issue the following command:

```
~# esxcli qlfastlinq qcc port iscsibootcfg settgtparams
```

Syntax

```
esxcli qlfastlinq qcc port iscsibootcfg settgtparams [cmd options]
```

Keywords

-l|--boot-lun <long> = Boot LUN
-d|--chap-id <str> = Target CHAP ID
-w|--chap-secret <str> = Target CHAP secret
-c|--connect <str> = Connect
-i|--identifier <str> = Port identifier (required)
-a|--ip-address <str> = Target IP address [a]
-q|--iscsi-name <str> = Target iSCSI name
-n|--target-number <long> = Target number (required)
-p|--tcp-port <long> = TCP port

[a] Addresses can contain either periods (`.`) or colons (`:`) to separate segments.

Parameter	Value (strings are case-sensitive)
`identifier`	Port identifier from the port `list` command
`chap-id`	Target CHAP IP string
`chap-secret`	Target CHAP password string
`ip-address`	Target IP address [a]
`iscsi-name`	Target IQN string
`target-number`	Target number whose parameter to be set (`0-1`)
`connect`	Enabled, Disabled
`boot-lun`	Boot LUN (`0-65535`)

[a] Addresses can contain either periods (`.`) or colons (`:`) to separate segments.

NOTE

To set target parameters, the target *must* be enabled (`connect=Enabled`).

Example

The following example shows the `port iscsibootcfg settgtparams` command on a local host:

```
~# esxcli qlfastlinq qcc port iscsibootcfg settgtparams -i
Port_27_0_0 -n 1 -c Enabled
```

The following example shows the `port iscsibootcfg settgtparams` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
iscsibootcfg settgtparams -i Port_27_0_0 -n1 -c Enabled
```

The output for both commands is:

```
iSCSI Boot Configuration:
   ISCSIBoot Info:
         Boot Targets:
         Connect: Enabled
         Target Boot LUN: 5
         Target Chap Id:
         Target Chap Secret:
         Target IPAddress: 192.168.100.227
         Target Iscsi Name:
         Target Number: 0
```

```
        Target Tcp Port: 3260

        Connect: Enabled
        Target Boot LUN: 0
        Target Chap Id:
        Target Chap Secret:
        Target IPAddress: 0.0.0.0
        Target Iscsi Name:
        Target Number: 1
        Target Tcp Port: 3260
        General Parameters:
        Boot To Iscsi Target: Enabled
        Chap Authentication: Disabled
        Dhcp Vendor Id: BRCM
        Hba Boot Mode: Disabled
        Ip Version: IPv4
        Iscsi Parameters Via DHCP: Disabled
        Link Up Delay Time: 55
        Lun Busy Retry Count: 0
        Target As First HDD: Disabled
        Tcp Ip Parameters Via DHCP: Enabled
        Use Tcp Timestamp: Disabled
        Initiator Parameters:
        Default Gateway: 0.0.0.0
        Initiator Chap Id:
        Initiator Chap Secret:
        Initiator IPAddress: 192.168.100.225
        Initiator Iscsi Name: iqn.1995-05.com.broadcom.iscsiboot
        Primary DNS: 0.0.0.0
        Secondary DNS: 0.0.0.0
        Subnet Mask: 0.0.0.0
        Mpio Parameters:
        MACAddress: 00:00:00:00:00:00
        Mpio Mode: Enabled
        Use Independent Target Name: Disabled
        Use Independent Target Portal: Enabled
  Message: Success
```

The output displayed is combination of existing settings and user modified settings saved in the cache file. The same data is present in the cache file.

Set iSCSI Boot Configuration MPIO Parameters

To save user-provided iSCSI boot configuration MPIO parameters information for a specified Marvell Adapter (3400/8400 Series) port to a temporary cache file, issue the following command:

```
~# esxcli qlfastlinq qcc port iscsibootcfg setmpioparams
```

Syntax

```
esxcli qlfastlinq qcc port iscsibootcfg setmpioparams [cmd
options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required)

`-n|--ind-target-name <str>` = Use independent target name

`-p|--ind-target-portal <str>` = Use independent target portal

`-a|--mpio-mac-address <str>` = MAC address [a]

`-m|--mpio-mode <str>` = MPIO mode

[a] Address must use colons (**:**) to separate segments.

Parameter	Value (strings are case-sensitive)
`identifier`	Port identifier from the port `list` command
`mpio-mode`	Enabled, Disabled
`ind-target-portal`	Enabled, Disabled
`ind-target-name`	Enabled, Disabled
`mpio-mac-address`	MPIO MAC address [a]

[a] Address must use colons (**:**) to separate segments.

NOTE

To set MPIO parameters, the MPIO mode *must* be enabled (`mpio-mode=Enabled`).

Example

The following example shows the `port iscsibootcfg setmpioparams` command on a local host:

```
~# esxcli qlfastlinq qcc port iscsibootcfg setmpioparams -i
Port_27_0_0 -m Enabled
```

The following example shows the `port iscsibootcfg setmpioparams` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
iscsibootcfg setmpioparams -i Port_27_0_0 -m Enabled
```

The output for both commands is:

```
iSCSI Boot Configuration:
   ISCSIBoot Info:
         Boot Targets:
         Connect: Enabled
         Target Boot LUN: 5
         Target Chap Id:
         Target Chap Secret:
         Target IPAddress: 192.168.100.227
         Target Iscsi Name:
         Target Number: 0
         Target Tcp Port: 3260

         Connect: Enabled
         Target Boot LUN: 0
         Target Chap Id:
         Target Chap Secret:
         Target IPAddress: 0.0.0.0
         Target Iscsi Name:
         Target Number: 1
         Target Tcp Port: 3260
         General Parameters:
         Boot To Iscsi Target: Enabled
         Chap Authentication: Disabled
         Dhcp Vendor Id: BRCM
         Hba Boot Mode: Disabled
         Ip Version: IPv4
         Iscsi Parameters Via DHCP: Disabled
         Link Up Delay Time: 55
         Lun Busy Retry Count: 0
         Target As First HDD: Disabled
         Tcp Ip Parameters Via DHCP: Enabled
         Use Tcp Timestamp: Disabled
         Initiator Parameters:
         Default Gateway: 0.0.0.0
```

```
            Initiator Chap Id:
            Initiator Chap Secret:
            Initiator IPAddress: 192.168.100.225
            Initiator Iscsi Name: iqn.1995-05.com.broadcom.iscsiboot
            Primary DNS: 0.0.0.0
            Secondary DNS: 0.0.0.0
            Subnet Mask: 0.0.0.0
            Mpio Parameters:
            MACAddress: 00:00:00:00:00:00
            Mpio Mode: Enabled
            Use Independent Target Name: Disabled
            Use Independent Target Portal: Enabled
   Message: Success
```

The command output contains a combination of existing settings and user-modified settings saved in the cache file.

Set iSCSI Boot Configuration

To import user-provided iSCSI boot configuration information for a specified Marvell Adapter (3400/8400 Series) port from a temporary cache file to flash, issue the following command:

```
~# esxcli qlfastlinq qcc port iscsibootcfg set
```

The `iscsibootcfg set` command fails if a cache file is not present. On successful completion of this command, the cache file is deleted.

Syntax

```
esxcli qlfastlinq qcc port iscsibootcfg set [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port iscsibootcfg set` command on a local host:

```
~# esxcli qlfastlinq qcc port iscsibootcfg set -i Port_27_0_0
```

The following example shows the `port iscsibootcfg set` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
iscsibootcfg set -i Port_27_0_0
```

The output for both commands is:

```
iSCSI Boot Configuration:
   ISCSIBoot Info:
         Boot Targets:
         Connect: Enabled
         Target Boot LUN: 5
         Target Chap Id:
         Target Chap Secret:
         Target IPAddress: 192.168.100.227
         Target Iscsi Name:
         Target Number: 0
         Target Tcp Port: 3260

         Connect: Disabled
         Target Boot LUN: 0
         Target Chap Id:
         Target Chap Secret:
         Target IPAddress: 0.0.0.0
         Target Iscsi Name:
         Target Number: 1
         Target Tcp Port: 0
         General Parameters:
         Boot To Iscsi Target: Enabled
         Chap Authentication: Disabled
         Dhcp Vendor Id: BRCM
         Hba Boot Mode: Disabled
         Ip Version: IPv4
         Iscsi Parameters Via DHCP: Disabled
         Link Up Delay Time: 0
         Lun Busy Retry Count: 0
         Target As First HDD: Disabled
         Tcp Ip Parameters Via DHCP: Enabled
         Use Tcp Timestamp: Disabled
         Initiator Parameters:
         Default Gateway: 0.0.0.0
         Initiator Chap Id:
         Initiator Chap Secret:
         Initiator IPAddress: 192.168.100.225
         Initiator Iscsi Name: iqn.1995-05.com.broadcom.iscsiboot
         Primary DNS: 0.0.0.0
```

```
            Secondary DNS: 0.0.0.0
            Subnet Mask: 0.0.0.0
            Mpio Parameters:
            MACAddress: 00:00:00:00:00:00
            Mpio Mode: Enabled
            Use Independent Target Name: Disabled
            Use Independent Target Portal: Enabled
      Message: Success
      Reboot Required: true
```

The output shows the new setting, which will be active after reboot if command is successful.

Adapter Port iSCSI Boot Configuration (41000/45000 Series)

The following command performs iSCSI boot configuration operations on 41000/45000 Series adapters:

```
~# esxcli qlfastlinq qcc port iscsibootcfg579xx
```

Syntax

```
esxcli qlfastlinq qcc port iscsibootcfg579xx {cmd} [cmd options]
```

Available Commands

`get`
Shows iSCSI boot configuration information for a specified Marvell Adapter (41000/45000 Series) port. See Get iSCSI Boot Configuration.

`setgenparams`
Saves iSCSI boot configuration general parameters in a file for a specified Marvell Adapter (41000/45000 Series) port. See Set iSCSI Boot Configuration General Parameters.

`setinitparams`
Saves iSCSI boot configuration initiator parameters in a file for a specified Marvell Adapter (41000/45000 Series) port. See Set iSCSI Boot Configuration Initiator Parameters.

`settgtparams`
Saves iSCSI boot configuration target parameters in a file for a specified Marvell Adapter (41000/45000 Series) port. See Set iSCSI Boot Configuration Target Parameters.

`set`
Imports iSCSI boot configuration information for a specified Marvell Adapter (41000/45000 Series) port. See Set iSCSI Boot Configuration.

Get iSCSI Boot Configuration

To show iSCSI boot configuration information for a specified Marvell Adapter (41000/45000 Series) port, issue the following command:

```
~# esxcli qlfastlinq qcc port iscsibootcfg579xx get
```

Syntax

```
esxcli qlfastlinq qcc port iscsibootcfg579xx get [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port iscsibootcfg579xx get` command on a local host:

```
~# esxcli qlfastlinq qcc port iscsibootcfg579xx get -i Port_17_0_0
```

The following example shows the `port iscsibootcfg579xx get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
iscsibootcfg579xx get -i Port_17_0_0
```

The output for both commands is:

```
iSCSI Boot Configuration:
   ISCSIBoot Info:
         Boot Targets:
         Connect: Enabled
         Target Boot LUN: 0
         Target Chap Id:
         Target Chap Secret:
         Target IPAddress: 0.0.0.0
         Target Iscsi Name:
         Target Number: 0
         Target Tcp Port: 3260

         Connect: Enabled
         Target Boot LUN: 0
         Target Chap Id:
         Target Chap Secret:
         Target IPAddress: 0.0.0.0
         Target Iscsi Name:
```

```
            Target Number: 1
            Target Tcp Port: 3260
            General Parameters:
            Address Redirect: Disabled
            Boot Mode: Non-Offload
            Boot Protocol: None
            Chap Authentication: Disabled
            Dhcp Request Timeout: 60
            Dhcp Vendor Id: QLGC
            IPv4Fallback: Disabled
            Ip Version: IPv4
            Iscsi Parameters Via DHCP: Enabled
            Mutual Chap: Disabled
            Target Login Timeout: 60
            Tcp Ip Parameters Via DHCP: Enabled
            Vlan: Enabled
            Vlan Id: 5976
            Initiator Parameters:
            Default Gateway: 0.0.0.0
            Initiator Chap Id:
            Initiator Chap Secret:
            Initiator IPAddress: 0.0.0.0
            Initiator Iscsi Name:
iqn.1994-02.com.qlogic.iscsi:fastlinqboot
            Primary DNS: 0.0.0.0
            Secondary DNS: 0.0.0.0
            Subnet Mask: 0.0.0.0
   Message: Success
```

Set iSCSI Boot Configuration General Parameters

To save user-provided iSCSI boot configuration general parameters information for a specified Marvell Adapter (41000/45000 Series) port to a temporary cache file, issue the following command:

```
~# esxcli qlfastlinq qcc port iscsibootcfg579xx setgenparams
```

Syntax

```
esxcli qlfastlinq qcc port iscsibootcfg579xx setgenparams [cmd
options]
```

Keywords

-e|--address-redirect <str> = Address redirect
-m|--boot-mode <str> = Boot mode
-o|--boot-protocol <str> = Boot protocol
-a|--chap-authentication <str> = CHAP authentication
-r|--dhcp-request-timeout <long> = DHCP request timeout
-d|--dhcp-vendor-id <str> = DHCP vendor ID
-i|--identifier <str> = Port identifier (required)
-f|--ipv4-fallback <str> = IPv4 fallback
-v|--ip-version <str> = IP version
-s|--iscsi-via-dhcp <str> = iSCSI parameters via DHCP
-u|--mutual-chap <str> = Mutual CHAP
-g|--target-login-timeout <long> = Target login timeout
-c|--tcpip-via-dhcp <str> = TCP/IP parameters via DHCP
-n|--vlan <str> = VLAN
-l|--vlan-id <long> = VLAN ID

Parameter	Value (strings are case-sensitive)
identifier	Port identifier from the port list command
address-redirect	Enabled, Disabled
chap-authentication	Enabled, Disabled
ipv4-fallback	Enabled, Disabled
iscsi-via-dhcp	Enabled, Disabled
mutual-chap	Enabled, Disabled
tcpip-via-dhcp	Enabled, Disabled
vlan	Enabled, Disabled
boot-mode	Offload, Non-Offload
boot-protocol	iSCSI, None
dhcp-vendor-id	String value

Parameter	Value (strings are case-sensitive)
ip-version	IPv4, IPv6
dhcp-request-timeout	Integer value (0–65535)
target-login-timeout	Integer value (0–65535)
vlan-id	Integer value (0–4094)

Example

The following example shows the `port iscsibootcfg579xx setgenparams` command on a local host:

```
~# esxcli qlfastlinq qcc port iscsibootcfg579xx setgenparams -i
Port_17_0_0 -e Disabled
```

The following example shows the `port iscsibootcfg579xx setgenparams` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
iscsibootcfg579xx setgenparams -i Port_17_0_0 -e Disabled
```

The output for both commands is:

```
iSCSI Boot Configuration:
   ISCSIBoot Info:
         Boot Targets:
         Connect: Enabled
         Target Boot LUN: 0
         Target Chap Id:
         Target Chap Secret:
         Target IPAddress: 0.0.0.0
         Target Iscsi Name:
         Target Number: 0
         Target Tcp Port: 3260

         Connect: Enabled
         Target Boot LUN: 0
         Target Chap Id:
         Target Chap Secret:
         Target IPAddress: 0.0.0.0
         Target Iscsi Name:
         Target Number: 1
         Target Tcp Port: 3260
         General Parameters:
         Address Redirect: Disabled
```

```
            Boot Mode: Non-Offload
            Boot Protocol: None
            Chap Authentication: Disabled
            Dhcp Request Timeout: 60
            Dhcp Vendor Id: QLGC
            IPv4Fallback: Disabled
            Ip Version: IPv4
            Iscsi Parameters Via DHCP: Enabled
            Mutual Chap: Disabled
            Target Login Timeout: 60
            Tcp Ip Parameters Via DHCP: Enabled
            Vlan: Enabled
            Vlan Id: 5976
            Initiator Parameters:
            Default Gateway: 0.0.0.0
            Initiator Chap Id:
            Initiator Chap Secret:
            Initiator IPAddress: 0.0.0.0
            Initiator Iscsi Name:
iqn.1994-02.com.qlogic.iscsi:fastlinqboot
            Primary DNS: 0.0.0.0
            Secondary DNS: 0.0.0.0
            Subnet Mask: 0.0.0.0
   Message: Success
```

The command output contains a combination of existing settings and user-modified settings saved in the cache file.

Set iSCSI Boot Configuration Initiator Parameters

To export user-provided iSCSI boot configuration initiator parameters information for a specified Marvell Adapter (41000/45000 Series) port from a temporary cache file, issue the following command:

```
~# esxcli qlfastlinq qcc port iscsibootcfg579xx setinitparams
```

Syntax

```
esxcli qlfastlinq qcc port iscsibootcfg579xx setinitparams [cmd
options]
```

Keywords

`-c|--chap-id <str>` = Initiator CHAP ID

`-w|--chap-secret <str>` = Initiator CHAP secret

```
-g|--default-gateway <str> = Initiator gateway address [a]
     -i|--identifier <str> = Port identifier (required)
     -a|--ip-address <str> = Initiator IP address [a]
     -q|--iscsi-name <str> = Initiator iSCSI name
   -p|--primary-dns <str> = Primary DNS address [a]
 -d|--secondary-dns <str> = Secondary DNS address [a]
   -m|--subnet-mask <str> = Initiator subnet mask address [a]
```

[a] Addresses can contain either periods (`.`) or colons (`:`) to separate segments.

Parameter	Value (strings are case-sensitive)
`identifier`	Port identifier from the port `list` command
`chap-id`	CHAP ID string
`chap-secret`	CHAP password string
`ip-address`	Initiator IP address [a]
`subnet-mask`	Initiator subnet mask [a]
`default-gateway`	Initiator default gateway [a]
`iscsi-name`	Initiator iSCSI qualified name (IQN)
`primary-dns`	Primary DNS address [a]
`secondary-dns`	Secondary DNS address [a]

[a] Addresses can contain either periods (`.`) or colons (`:`) to separate segments.

Example

The following example shows the `port iscsibootcfg579xx setinitparams` command on a local host:

```
~# esxcli qlfastlinq qcc port iscsibootcfg579xx setinitparams -i
Port_17_0_0 -a 192.169.100.224
```

The following example shows the `port iscsibootcfg579xx setinitparams` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
iscsibootcfg579xx setinitparams -i Port_17_0_0 -a 192.168.100.224
```

The output for both commands is:

```
iSCSI Boot Configuration:
   ISCSIBoot Info:
         Boot Targets:
```

```
            Connect: Enabled
            Target Boot LUN: 0
            Target Chap Id:
            Target Chap Secret:
            Target IPAddress: 0.0.0.0
            Target Iscsi Name:
            Target Number: 0
            Target Tcp Port: 3260

            Connect: Enabled
            Target Boot LUN: 0
            Target Chap Id:
            Target Chap Secret:
            Target IPAddress: 0.0.0.0
            Target Iscsi Name:
            Target Number: 1
            Target Tcp Port: 3260
            General Parameters:
            Address Redirect: Disabled
            Boot Mode: Non-Offload
            Boot Protocol: None
            Chap Authentication: Disabled
            Dhcp Request Timeout: 60
            Dhcp Vendor Id: QLGC
            IPv4Fallback: Disabled
            Ip Version: IPv4
            Iscsi Parameters Via DHCP: Enabled
            Mutual Chap: Disabled
            Target Login Timeout: 60
            Tcp Ip Parameters Via DHCP: Enabled
            Vlan: Enabled
            Vlan Id: 5976
            Initiator Parameters:
            Default Gateway: 0.0.0.0
            Initiator Chap Id:
            Initiator Chap Secret:
            Initiator IPAddress: 192.169.100.224
            Initiator Iscsi Name:
iqn.1994-02.com.qlogic.iscsi:fastlinqboot
            Primary DNS: 0.0.0.0
```

```
        Secondary DNS: 0.0.0.0
        Subnet Mask: 0.0.0.0
  Message: Success
```

The command output contains a combination of existing settings and user-modified settings saved in the cache file.

Set iSCSI Boot Configuration Target Parameters

To save user-provided iSCSI boot configuration target parameters information for a specified Marvell Adapter (41000/45000 Series) port to a temporary cache, issue the following command:

```
~# esxcli qlfastlinq qcc port iscsibootcfg579xx settgtparams
```

Syntax

```
esxcli qlfastlinq qcc port iscsibootcfg579xx settgtparams [cmd options]
```

Keywords

```
      -l|--boot-lun <long> = Boot LUN
        -d|--chap-id <str> = Target CHAP ID
    -w|--chap-secret <str> = Target CHAP secret
        -c|--connect <str> = Connect
     -i|--identifier <str> = Port identifier (required)
     -a|--ip-address <str> = Target IP address [a]
     -q|--iscsi-name <str> = Target iSCSI name
-n|--target-number <long> = Target number (required)
      -p|--tcp-port <long> = TCP port
```

[a] Addresses can contain either periods (`.`) or colons (`:`) to separate segments.

Parameter	Values (strings are case-sensitive)
`identifier`	Port identifier from the port `list` command
`chap-id`	Target CHAP IP string
`chap-secret`	Target CHAP password string
`ip-address`	Target IP address [a]
`iscsi-name`	Target IQN name string

Parameter	Values (strings are case-sensitive)
`target-number`	Target number (`0-1`)
`tcp-port`	Target TCP Port (`0-65535`)
`connect`	Enabled, Disabled
`boot-lun`	Boot LUN number (`0-65535`)

[a] Addresses can contain either periods (`.`) or colons (`:`) to separate segments.

NOTE

To set target parameters, the target *must* be enabled (`connect=Enabled`).

Example

The following example shows the `port iscsibootcfg579xx settgtparams` command on a local host:

```
~# esxcli qlfastlinq qcc port iscsibootcfg579xx settgtparams -i
Port_17_0_0 -c Enabled -n 0
```

The following example shows the `port iscsibootcfg579xx settgtparams` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
iscsibootcfg579xx settgtparams -i Port_17_0_0 -c Enabled -n 0
```

The output for both commands is:

```
iSCSI Boot Configuration:
   ISCSIBoot Info:
         Boot Targets:
         Connect: Enabled
         Target Boot LUN: 0
         Target Chap Id:
         Target Chap Secret:
         Target IPAddress: 0.0.0.0
         Target Iscsi Name:
         Target Number: 0
         Target Tcp Port: 3260

         Connect: Enabled
         Target Boot LUN: 0
         Target Chap Id:
         Target Chap Secret:
```

```
            Target IPAddress: 0.0.0.0
            Target Iscsi Name:
            Target Number: 1
            Target Tcp Port: 3260
            General Parameters:
            Address Redirect: Disabled
            Boot Mode: Non-Offload
            Boot Protocol: None
            Chap Authentication: Disabled
            Dhcp Request Timeout: 60
            Dhcp Vendor Id: QLGC
            IPv4Fallback: Disabled
            Ip Version: IPv4
            Iscsi Parameters Via DHCP: Enabled
            Mutual Chap: Disabled
            Target Login Timeout: 60
            Tcp Ip Parameters Via DHCP: Enabled
            Vlan: Enabled
            Vlan Id: 5976
            Initiator Parameters:
            Default Gateway: 0.0.0.0
            Initiator Chap Id:
            Initiator Chap Secret:
            Initiator IPAddress: 192.169.100.224
            Initiator Iscsi Name:
iqn.1994-02.com.qlogic.iscsi:fastlinqboot
            Primary DNS: 0.0.0.0
            Secondary DNS: 0.0.0.0
            Subnet Mask: 0.0.0.0
   Message: Success
```

The command output contains a combination of existing settings and user-modified settings saved in the cache file.

Set iSCSI Boot Configuration

To import user-provided iSCSI boot configuration information for a specified Marvell Adapter (41000/45000 Series) port from a temporary cache file to flash, issue the following command:

```
~# esxcli qlfastlinq qcc port iscsibootcfg579xx set
```

The `set` command fails if the cache file is not present. On successful completion of the command, the cache file is deleted.

Syntax

```
esxcli qlfastlinq qcc port iscsibootcfg579xx set [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` command.

Example

The following example shows the `port iscsibootcfg set` command on a local host:

```
~# esxcli qlfastlinq qcc port iscsibootcfg579xx set -i Port_17_0_0
```

The following example shows the `port iscsibootcfg set` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
iscsibootcfg set -i Port_27_0_0
```

The output for both commands is:

```
iSCSI Boot Configuration:
   ISCSIBoot Info:
         Boot Targets:
         Connect: Enabled
         Target Boot LUN: 0
         Target Chap Id:
         Target Chap Secret:
         Target IPAddress: 0.0.0.0
         Target Iscsi Name:
         Target Number: 0
         Target Tcp Port: 3260

         Connect: Enabled
         Target Boot LUN: 0
         Target Chap Id:
         Target Chap Secret:
         Target IPAddress: 0.0.0.0
         Target Iscsi Name:
         Target Number: 1
         Target Tcp Port: 3260
         General Parameters:
         Address Redirect: Disabled
         Boot Mode: Non-Offload
         Boot Protocol: None
```

```
            Chap Authentication: Disabled
            Dhcp Request Timeout: 60
            Dhcp Vendor Id: QLGC
            IPv4Fallback: Disabled
            Ip Version: IPv4
            Iscsi Parameters Via DHCP: Enabled
            Mutual Chap: Disabled
            Target Login Timeout: 60
            Tcp Ip Parameters Via DHCP: Enabled
            Vlan: Enabled
            Vlan Id: 5976
            Initiator Parameters:
            Default Gateway: 0.0.0.0
            Initiator Chap Id:
            Initiator Chap Secret:
            Initiator IPAddress: 192.169.100.224
            Initiator Iscsi Name:
iqn.1994-02.com.qlogic.iscsi:fastlinqboot
            Primary DNS: 0.0.0.0
            Secondary DNS: 0.0.0.0
            Subnet Mask: 0.0.0.0
   Message: Success
   Reboot Required: true
```

The command output shows the new settings, which are active after reboot if command is successful.

Adapter Port MBA Boot Configuration Namespace Commands and Sub-namespaces

Use the commands in this section to perform Multi-Boot Agent (MBA) boot configuration operations on 3400/8400 Series and 41000/45000 Series Adapter ports.

```
~# esxcli qlfastlinq qcc port mbabootcfg
```

Syntax

```
esxcli qlfastlinq qcc port mbabootcfg {cmd} [cmd options]
```

Available Commands

`get`
Shows MBA boot configuration information about the specified Marvell Adapter port. See Get MBA Boot Configuration.

`set`
Sets MBA boot configuration information about the specified Marvell Adapter port. See Set MBA Boot Configuration.

Get MBA Boot Configuration

To show MBA boot configuration information for a specified 3400/8400 Series or 41000/45000 Series Adapter port, issue the following command:

```
~# esxcli qlfastlinq qcc port mbabootcfg get
```

Syntax

```
esxcli qlfastlinq qcc port mbabootcfg get [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` command.

Example

The following example shows the `port mbabootcfg get` command on a local host:

```
~# esxcli qlfastlinq qcc port mbabootcfg get -i Port_17_0_0
```

The following example shows the `port mbabootcfg get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
mbabootcfg get -i Port_17_0_0
```

The output for both commands is:

```
MBA Boot Configuration:
   MBABoot Info:
         Banner Message Timeout: 15
         Boot Protocol: None
         Boot Retry Count: 0
         Boot Strap Type: Auto
         Hide Setup Prompt: Disabled
         Link Speed: 40_Gbps
         Option ROM: Enabled
         Preboot Wakeon LAN: Disabled
         Setup Key Stroke: Ctrl-S
         VLANMode: Disabled
         VLANID: 1
   Message: Success
```

Set MBA Boot Configuration

To configure MBA boot information for a specified 3400/8400 Series or 41000/45000 Series Adapter port, issue the following command:

```
~# esxcli qlfastlinq qcc port mbabootcfg set
```

Syntax

```
esxcli qlfastlinq qcc port mbabootcfg set [cmd options]
```

Keywords

```
-n|--banner-msg-timeout <long> = Banner message timeout
      -p|--boot-protocol <str> = Boot protocol
  -c|--boot-retry-count <long> = Boot retry count
    -y|--boot-strap-type <str> = Boot strap type
  -e|--hide-setup-prompt <str> = Hide setup prompt
        -i|--identifier <str> = Port identifier (required)
        -s|--link-speed <str> = Link speed
        -o|--option-rom <str> = Option ROM
     -w|--pre-boot-wol <str> = Pre-boot wake on LAN
 -k|--setup-key-stroke <str> = Setup Key Stroke
         -d|--vlan-id <long> = VLAN ID
        -m|--vlan-mode <str> = VLAN mode
```

Parameter	Values (strings are case-sensitive)
`identifier`	Port identifier from the port `list` command
`option-rom`	Enabled, Disabled
`boot-protocol`	For 3400/8400 Series adapters: None, PXE, iBFT, iSCSI, FCoE, RPL For 41000/45000 Series adapters: None, PXE, iBFT
`boot-strap-type`	Int_19h, Int_18h, BBS, Auto
`hide-setup-prompt`	Enabled, Disabled
`setup-key-stroke`	Ctrl-B, Ctrl-S
`banner-message-timeout`	Integer value (`1-15`)
`link-speed`	For 3400/8400 Series adapters: 1_Gbps, 10_Gbps For 41000/45000 Series adapters: 10_Gbps, 25_Gbps, 40_Gbps, 50_Gbps, 100_Gbps, Auto_Neg

Parameter	Values (strings are case-sensitive)
pre-boot-wol	Enabled, Disabled
vlan-mode	Enabled, Disabled
vlan-id	Integer value (1–4094)
boot-retry-count	Integer value (1–64)

Example

The following example shows the `port mbabootcfg set` command on a local host:

```
~# esxcli qlfastlinq qcc port mbabootcfg set -i Port_17_0_0 -n 13
```

The following example shows the `port mbabootcfg set` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
mbabootcfg set -i Port_17_0_0 -n 13
```

The output for both commands is:

```
MBA Boot Configuration:
   MBABoot Info:
         Banner Message Timeout: 13
         Boot Protocol: None
         Boot Retry Count: 0
         Boot Strap Type: Auto
         Hide Setup Prompt: Disabled
         Link Speed: 40_Gbps
         Option ROM: Enabled
         Preboot Wakeon LAN: Disabled
         Setup Key Stroke: Ctrl-S
         VLANMode: Disabled
         VLANID: 1
   Message: Success
   Reboot Required: true
```

The output shows the new settings, which will be active after a reboot if the command is successful.

Adapter Port Diagnostics Namespace Commands and Sub-namespaces

Use the commands in this section to perform diagnostics on 3400/8400 Series and 41000/45000 Series Adapter ports.

```
~# esxcli qlfastlinq qcc port diag
```

Syntax

```
esxcli qlfastlinq qcc port diag {cmd} [cmd options]
```

Available Commands

`test`
Runs diagnostics tests on the identified Marvell Adapter port. See Run Port Diagnostics Test.

Run Port Diagnostics Test

To run port diagnostic tests for a specified Marvell Adapter 3400/8400 Series or 41000/45000 Series port, issue the following command:

```
~# esxcli qlfastlinq qcc port diag test
```

Syntax

```
esxcli qlfastlinq qcc port diag test [cmd options]
```

Keywords

`-n|--diag-test-name <str>` = Diagnostic test command (required)
`-i|--identifier <str>` = Port identifier (vmnic) (required). Obtain the identifier from the `port list` command output.
`-l|--led-interval <long>` = LED Interval (for LED test only)
`-c|--test-iterations <long>` = Number of test iterations (required)

Parameter	Values (strings are case-sensitive)
`identifier`	Port identifier from port list command
`diag-test-name`	ControlRegisters, InternalMemory, EEPROM, Interrupt, LoopbackPHY, LoopbackMAC, LED
`test-iterations`	Integer value (`1–255`)
`led-interval`	Integer value (`5–100` seconds) (default = `5`)

Example

The following example shows the `port diag test` command on a local host:

```
~# esxcli qlfastlinq qcc port diag test -i Port_17_0_0 -n LED -c 2
-l 10
```

The following example shows the `port diag test` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
diag test -i Port_17_0_0 -n LED -c 2 -l 10
```

The output for both commands is:

```
Diagnostic Test Result:
   Message: Success
   Test Name: LED
```

Adapter Port DCBX Namespace Commands and Sub-namespaces

The following command performs DCBX operations on 3400/8400 Series and 41000/45000 Series Adapters:

```
~# esxcli qlfastlinq qcc port dcbx
```

Syntax

```
esxcli qlfastlinq qcc port dcbx {cmd} [cmd options]
```

Available Commands

`get`
Shows DCBX information for the specified Marvell Adapter port. See Get DCBX Port Information.

Get DCBX Port Information

To show DCBX data for a specified 3400/8400 Series or 41000/45000 Series Adapter port, issue the following command:

```
~# esxcli qlfastlinq qcc port dcbx get
```

Syntax

```
esxcli qlfastlinq qcc port dcbx get [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the `port list` command output.

Example

The following example shows the `port dcbx get` command on a local host:

```
~# esxcli qlfastlinq qcc port dcbx get -i Port_132_0_0
```

The following example shows the `port dcbx get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
dcbx get -i Port_132_0_0
```

The output for both commands is:

```
DCBX Information:
   Dcbx Info:
         DCB: Enabled
         Dcb Protocol: Auto
         Dcbx Advanced:
         Local MIB:
         Configuration Mismatch: Disabled
         ETS: Disabled
         Fcoe PGID: NA
         Fcoe PRI: NA
         Iscsi PGID: NA
         Iscsi PRI: NA
         Networking PGID: 0
         Networking PRI: 0
         PFC: Disabled
         PGID_0_BW: 0
         PGID_1_BW: 0
         PGID_2_BW: 0
         PGID_3_BW: 0
         PGID_4_BW: 0
         PGID_5_BW: 0
         PGID_6_BW: 0
         PGID_7_BW: 0
         Pfc Disabled On Priorities: 0 1 2 3 4 5 6 7
         Pfc Enabled On Priorities:
         Remote MIB:
         Remote Application Priority Willing: Enabled
         Remote Ets Recomendation Valid: Disabled
         Remote Ets Willing: Enabled
         Remote Fcoe PGID: NA
         Remote Fcoe PRI: NA
```

```
            Remote Iscsi PGID: NA
            Remote Iscsi PRI: NA
            Remote Networking PGID: 0
            Remote PGID_0_BW: 50
            Remote PGID_1_BW: 50
            Remote PGID_2_BW: 0
            Remote PGID_3_BW: 0
            Remote PGID_4_BW: 0
            Remote PGID_5_BW: 0
            Remote PGID_6_BW: 0
            Remote PGID_7_BW: 0
            Remote Pfc Disabled On Priorities: 0 1 2 4 5 6 7
            Remote Pfc Enabled On Priorities: 3
            Remote Pfc Willing: Enabled
            Enhanced Transmission Selection: Not Operational
            Priority Flow Control: Not Operational
            Priority Flow Control Values:
            Priority Group Bandwidth:
            Priority Tagging: Not Operational
            Priority Tagging Values:
            Priority To Priority Group Mapping:
   Message: Success
```

Adapter Port Link Settings (SmartAN) Namespace Commands and Sub-namespaces

The following command performs link setting (Marvell Smart autonegotiation (SmartAN™)) operations on the 41000/45000 Series Adapters:

```
~# esxcli qlfastlinq qcc port linkSettings
```

Syntax

```
esxcli qlfastlinq qcc port linkSettings {cmd} [cmd options]
```

Available Commands

`get`
Shows link settings (SmartAN) information for the specified Marvell Adapter port. See Get Port Link Settings Information.

`set`
Configures link settings (SmartAN) information for the specified Marvell Adapter port. See Set Port Link Settings Information.

Get Port Link Settings Information

To view port link settings (SmartAN) for a specified 41000/45000 Series adapter port, issue the following command:

```
~# esxcli qlfastlinq qcc port linkSettings get
```

Syntax

```
esxcli qlfastlinq qcc port linkSettings get [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the `port list` command output.

Example

The following example shows the `port linkSettings get` command on a local host:

```
~# esxcli qlfastlinq qcc port linkSettings get -i Port_13_0_0
```

The following example shows the `port linkSettings get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
linkSettings get -i Port_13_0_0
```

The output for both commands is:

```
LinkSettings:
   Link Settings Info:
         Current Value: SmartAN
         Description: SmartAN(TM) Mode
         Name: SmartAN
         Supported Values: SmartAN,Advanced_Link_Settings
   Message: Success
```

The output depends upon the current mode of the port link settings.

Set Port Link Settings Information

To configure port link settings (SmartAN) for a specified 41000/45000 Series Adapter port, issue the following command:

```
~# esxcli qlfastlinq qcc port linkSettings set
```

Syntax

```
esxcli qlfastlinq qcc port linkSettings set [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (vmnic) (required). Obtain the identifier from the `port list` command output.

`-n|--name=<str>` = Parameter name (required)

`-v|--value=<str>` = Parameter value from list of supported values provided in the `get` command (required)

Example

The following example shows the `port linkSettings set` command on a local host:

```
~# esxcli qlfastlinq qcc port linkSettings set -i Port_13_0_0 –n
SmartAN –v Advanced_Link_Settings
```

The following example shows the `port linkSettings set` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
linkSettings set -i Port_13_0_0 –n SmartAN –v
Advanced_Link_Settings
```

The output for both commands is:

```
LinkSettings:
   Link Settings Info:
         Current Value: Advanced_Link_Settings
         Description: SmartAN(TM) Mode
         Name: SmartAN
         Supported Values: SmartAN,Advanced_Link_Settings

         Current Value: AutoNeg
         Description: NVM Speed
         Name: NVM
         Supported Values: Unknown,AutoNeg,1G,10G

         Current Value: None
         Description: FEC An Mode
         Name: FECAnMode
         Supported Values: None,10G_Firecode,ALL
   Message: Success
   Reboot Required: true
```

The new settings will be active after reboot.

The output depends upon current mode of the port link settings.

The `linkSettings` are on a per-port basis for the 41000 Series Adapter. For 45000 Series Adapters, the same settings are written on all ports of the same adapter.

Adapter Port DMI Namespace Commands and Sub-namespaces

The port desktop management interface (DMI) command displays small form-factor pluggable (SFP) DMI information for the 41000/45000 Series SFP-based adapters when optics/DAC is connected. This command is not supported on BASE-T interfaced 41000/45000 Series adapters nor on any 3400/8400 (578*xx*) Series adapters.

```
~# esxcli qlfastlinq qcc port dmi
```

Syntax

```
esxcli qlfastlinq qcc port dmi {cmd} [cmd options]
```

Available Commands

`get`
Shows SFP DMI information for the specified Marvell Adapter port. See Get Port Link Settings Information.

Get Port DMI Information

To view SFP DMI data for the adapters, issue the following command:

```
~# esxcli qlfastlinq qcc port dmi get
```

Syntax

```
esxcli qlfastlinq qcc port dmi get [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the `port list` command output.

Example

The following example shows the `port dmi get` command on a local host:

```
~# esxcli qlfastlinq qcc port dmi get -i Port_8_0_0
```

The following example shows the `port dmi get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc port
dmi get -i Port_8_0_0
```

The output for both commands is:

```
DMI Information:
   Dmi Info:
         Dmi Data:

------------------------------------------------------------------------------
Media Information
------------------------------------------------------------------------------
              Vendor: Amphenol
                Type: 100Gbit/Secor25Gbit/Sec
         Part Number: NDAQGJ-0005
               Speed: 100 Gbit/Sec or 25 Gbit/Sec
            Revision: D
       Serial Number: APF16200057E0V

------------------------------------------------------------------------------

Optical Transceiver Digital Diagnostic Data:

Address A0
           Identifier: SFP
       Ext Identifier: GBIC/SFP defined by serial ID only
            Connector:
       Ethernet Speed:
           Compliance: 0x00 0x00 0x00
       FC Link Length:
  FC Transmitter Tech: Copper Passive (IR1/IR2) Electrical Intra-Enclosure (EL)
FC Transmission Media:
             FC Speed:
             Encoding:
          BR, Nominal: 0xff
    Length (9um) - km: 0x00
         Length (9um): 0x00
        Length (50um): 0x00
      Length (62.5um): 0x00
      Length (Copper): 0x05
          Vendor name: Amphenol
   Ext Ethernet Speed: 100 Gbit/Sec or 25 Gbit/Sec
           Vendor OUI: 0x78 0xa7 0x14
```

```
            Vendor PN: NDAQGJ-0005
           Vendor rev: D
           Wavelength: 0x0100
              CC_BASE: 0x17
              Options:-Signal Loss, as defined in SFP MSA: 0x0
                      -Signal Loss, inverted from SFP MSA: 0x0
                      -TX_FAULT signal implemented: 0x0
                      -TX_DISABLE implemented and disables serial o/p: 0x0
                      -RATE_SELECT implemented: 0x0
              BR, max: 0x67
              BR, min: 0x00
            Vendor SN: APF16200057E0V
            Date code: 160520
 Diag Monitoring Type:-Address change required: 0x0
                      -Power Measurement: 0x0
                      -Externally Calibrated: 0x0
                      -Internally Calibrated: 0x0
                      -Digital diag monitoring: 0x0
                      -Legacy diagnostic: 0x0
     Enhanced Options:-Soft RATE_SELECT ctrl and monitoring: 0x0
                      -Soft RX_LOS monitoring: 0x0
                      -Soft TX_FAULT monitoring: 0x0
                      -Soft TX_DISABLE ctrl and monitoring: 0x0
                      -Alarm/warning flags: 0x0
  SFF-8472 Compliance: Digital diag not included or undefined
               CC_EXT: 0x4c
      Vendor Specific: 0x41 0xff 0xff 0xff 0xff 0xff 0xff 0xff
                       0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff
                       0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff
                       0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff

Address A2
       Temp High Alarm: 0x0000
        Temp Low Alarm: 0x0000
     Temp High Warning: 0x0000
      Temp Low Warning: 0x0000
    Voltage High Alarm: 0x0000
     Voltage Low Alarm: 0x0000
  Voltage High Warning: 0x0000
   Voltage Low Warning: 0x0000
```

```
      Bias High Alarm: 0x0000
       Bias Low Alarm: 0x0000
    Bias High Warning: 0x0000
     Bias Low Warning: 0x0000
TX Signal Power High Alarm: 0x0000
TX Signal Power Low Alarm: 0x0000
TX Signal Power High Warning: 0x0000
TX Signal Power Low Warning: 0x0000
RX Signal Power High Alarm: 0x0000
RX Signal Power Low Alarm: 0x0000
RX Signal Power High Warning: 0x0000
RX Signal Power Low Warning: 0x0000
           Rx_PWR(4): 0x00000000
           Rx_PWR(3): 0x00000000
           Rx_PWR(2): 0x00000000
           Rx_PWR(1): 0x00000000
           Rx_PWR(0): 0x00000000
         Tx_I(Slope): 0x0000
        Tx_I(Offset): 0x0000
       Tx_PWR(Slope): 0x0000
      Tx_PWR(Offset): 0x0000
            T(Slope): 0x0000
           T(Offset): 0x0000
            V(Slope): 0x0000
           V(Offset): 0x0000
            Checksum: 0x00
      Temperature MSB: 0x00
      Temperature LSB: 0x00
              Vcc MSB: 0x00
              Vcc LSB: 0x00
          TX Bias MSB: 0x00
          TX Bias LSB: 0x00
         TX Power MSB: 0x00
         TX Power LSB: 0x00
         RX Power MSB: 0x00
         RX Power LSB: 0x00
         Reserved MSB: 0x00
         Reserved LSB: 0x00
         Reserved MSB: 0x00
```

```
        Reserved LSB: 0x00
  Status/Control Bits:-Data_Ready_Bar: 0x0
                      -LOS: 0x0
                      -TX Fault: 0x0
                      -Soft RX Rate Select: 0x0
                      -RX Rate Select State: 0x0
                      -Soft TX Disable: 0x0
                      -TX Disable State: 0x0
                      -TX Signal Power Low Alarm: 0x0
                      -TX Signal Power High Alarm: 0x0
                      -TX Bias Low Alarm: 0x0
                      -TX Bias High Alarm: 0x0
                      -Vcc Low Alarm: 0x0
                      -Vcc High Alarm: 0x0
                      -Temp Low Alarm: 0x0
                      -Temp High Alarm: 0x0
                      -RX Signal Power Low Alarm: 0x0
                      -RX Signal Power High Alarm: 0x0
                      -TX Signal Power Low Warning: 0x0
                      -TX Signal Power High Warning: 0x0
                      -TX Bias Low Warning: 0x0
                      -TX Bias High Warning: 0x0
                      -Vcc Low Warning: 0x0
                      -Vcc High Warning: 0x0
                      -Temp Low Warning: 0x0
                      -Temp High Warning: 0x0
                      -Reserved Warning: 0x0
                      -Reserved Warning: 0x0
                      -Reserved Warning: 0x0
                      -Reserved Warning: 0x0
                      -Reserved Warning: 0x0
                      -Reserved Warning: 0x0
                      -RX Signal Power Low Warning: 0x0
                      -RX Signal Power High Warning: 0x0
     Vendor Specific: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
   Message: Success
```

Adapter NIC Namespace Commands and Sub-namespaces

To view NIC namespace and sub-namespace commands, issue the following:

```
~# esxcli qlfastlinq qcc nic
```

Syntax

```
esxcli qlfastlinq qcc nic {cmd} [cmd options]
```

Namespaces

Available namespaces include the following:

Namespace	Description
`advParams`	QCC CLI for NIC advanced parameters
`info`	QCC CLI for operations on NIC information
`stats`	QCC CLI for NIC statistics

Available Commands

`list`
Lists identifiers for the Marvell Adapter NICs discovered on the host. See List NIC Ports.

List NIC Ports

The `nic list` command lists all supported 3400/8400 Series and 41000/45000 Series NIC ports.

Example

The following example shows the `nic list` command on a local host:

```
~# esxcli qlfastlinq qcc nic list
```

The following example shows the `nic list` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc nic
list
```

The output for both commands is:

```
Name   PCI Device   Driver  Link Status  Speed  Duplex  MAC Address       MTU    Description
----   ----------   ------- -----------  -----  ------ ----------        -----  -------------
vmnic8  0000:11:00.0  qedentv  Down         0    Full  00:0E:1E:C4:B2:D5  1500  FastLinQ QL45462H 40GbE Adapter
                                                                                  rev10 (vmnic8)
vmnic9  0000:11:00.1  qedentv  Up       40000    Full  00:0E:1E:C4:B2:D6  1500  FastLinQ QL45462H 40GbE Adapter
                                                                                  rev 10 (vmnic9)
vmnic4  0000:1b:00.0  bnx2x    Down         0    Half  00:0A:F7:73:74:10  1500  BCM57800 1/10 Gigabit Ethernet
                                                                                  rev 10 (vmnic4)
vmnic5  0000:1b:00.1  bnx2x    Down         0    Half  00:0A:F7:73:74:11  1500  BCM57800 1/10 Gigabit Ethernet
                                                                                  rev 10 (vmnic5)
vmnic6  0000:1b:00.2  bnx2x    Down            0  Half  00:0A:F7:73:74:12  1500  BCM57800 1/10 Gigabit Ethernet
                                                                                  rev 10 (vmnic6)
```

```
vmnic7  0000:1b:00.3  bnx2x    Down           0  Half  00:0A:F7:73:74:13  1500  BCM57800 1/10 Gigabit Ethernet
                                                                          rev 10 (vmnic7)
```

Adapter NIC Information

```
~# esxcli qlfastlinq qcc nic info
```

Syntax

```
esxcli qlfastlinq qcc nic info {cmd} [cmd options]
```

Available Commands

`get`
Shows information about the specified Marvell Adapter NIC function. See Get NIC Information.

Get NIC Information

To view NIC information for the selected NIC device, issue the following command:

```
~# esxcli qlfastlinq qcc nic info get
```

Syntax

```
esxcli qlfastlinq qcc nic info get [cmd options]
```

Keywords

`-i|--identifier <str>` = NIC identifier (vmnic) (required). Obtain the identifier from the `nic list` command output.

Example

The following example shows the `nic info get` command on a local host:

```
~# esxcli qlfastlinq qcc nic info get -i vmnic9
```

The following example shows the `nic info get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc nic
info get -i vmnic9
```

The output for both commands is:

```
NIC Information:
   Message: Success
   NICInfo:
         Driver Date: NA
         Driver Name: qedentv
         Driver Version: 3.0.6.6
         Flow Control: Rx/Tx
         Interface: Up
```

```
Link Speed In Mbps: 40000
MACAddress: 00:0E:1E:C4:B2:D6
MTU: 1500
Permanent MACAddress: 00:0E:1E:C4:B2:D6
```

Adapter NIC Statistics

```
~# esxcli qlfastlinq qcc nic stats
```

Syntax

```
esxcli qlfastlinq qcc nic stats {cmd} [cmd options]
```

Available Commands

`get`
Shows statistics for the specified Marvell Adapter NIC function. Get NIC Statistics.

Get NIC Statistics

To view NIC statistics for the specified Marvell Adapter NIC function, issue the following command:

```
~# esxcli qlfastlinq qcc nic stats get
```

Syntax

```
esxcli qlfastlinq qcc nic stats get [cmd options]
```

Keywords

`-i|--identifier <str>` = NIC identifier (vmnic name) (required). Obtain the identifier from the `nic list` command output.

Example

The following example shows the `nic stats get` command on a local host:

```
~# esxcli qlfastlinq qcc nic stats get -i vmnic9
```

The following example shows the `nic stats get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc nic
stats get -i vmnic9
```

The output for both commands is:

```
NIC Statistics:
   Message: Success
   NICStats:
         Broadcast Frames Received: 0
         Broadcast Frames Transmitted: 0
         Carrier Sense Errors: 0
```

```
Deferred Transmissions: 0
Directed Frames Received: 0
Directed Frames Transmitted: 0
Excessive Collisions: 0
Fec Core Error: 0
Fec Mode: 0
Fec Reserved: 0
Fec Uncore Error: 0
Late Collisions: 0
Multicast Frames Received: 0
Multicast Frames Transmitted: 0
Multiple Collision Frames: 0
Octets Received: 16470171
Octets Transmitted: 0
Packets Received: 0
Packets Transmitted: 0
Receive Threshold Hits: 0
Single Collision Frames: 0
Total Receive Errors: 0
Total Transmit Errors: 0
Transmit Threshold Hits: 0
```

Adapter NIC Advanced Parameters Namespace Commands and Sub-namespaces

The following command performs NIC advanced parameters operations on 3400/8400 Series Adapters and 41000/45000 Series Adapters:

```
~# esxcli qlfastlinq qcc nic advParams
```

Syntax

```
esxcli qlfastlinq qcc advParams {cmd} [cmd options]
```

Available Commands

`get`
Shows NIC advanced parameter information for the specified Marvell Adapter port. See Get NIC Advanced Parameter Information.

`set`
Configures NIC advanced parameter information for the specified Marvell Adapter port. See Set NIC Advanced Parameter Information.

Get NIC Advanced Parameter Information

To view port NIC advanced parameter information for a specified 3400/8400/41000/45000 Series Adapter, issue the following command:

```
~# esxcli qlfastlinq qcc advParams get
```

Syntax

```
esxcli qlfastlinq qcc advParams get [cmd options]
```

Keywords

`-i|--identifier <str>` = NIC identifier (vmnic name) (required). Obtain the identifier from the `nic list` command output.

Example

The following example shows the `nic advParams get` command on a local host:

```
~# esxcli qlfastlinq qcc nic advParams get -i vmnic0
```

The following example shows the `nic advParams get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc nic
advParams get -i vmnic0
```

The output for both commands is:

```
Advanced Params:
   Adv Params:
         Current Value: Disabled
         Default Value: Disabled
         Description: RoCE
         Name: RoCE
         Param Type: String-Enum
         String Max Length: NA
         Supported Values: Disabled

         Current Value: 0
         Default Value: 0
         Description: RoCE Current MTU
         Name: RoCECurrMTU
         Param Type: String-Enum
         String Max Length: NA
         Supported Values: 0
   Message: Success
```

The output depends upon number of parameters supported by driver.

Set NIC Advanced Parameter Information

To configure NIC advanced parameters for a specified 3400/8400 Series adapter, issue the following command:

```
~# esxcli qlfastlinq qcc nic advParams set
```

Syntax

```
esxcli qlfastlinq qcc nic advParams set [cmd options]
```

Keywords

`-i|--identifier <str>` = NIC identifier (vmnic name) (required). Obtain the identifier from the `nic list` command output.

`-n|--name=<str>` = Parameter name (required)

`-v|--value=<str>` = Parameter value from list of supported values provided in the `get` command (required)

Example

The following example shows the `nic advParms set` command on a local host:

```
~# esxcli qlfastlinq qcc nic advParams set -i vmnic14 –n NvmDcbxCFg
–v Enabled
```

The following example shows the `nic advParams set` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc nic
advParams set -i vmnic14 –n NvmDcbxCFg –v Enabled
```

The output for both commands is:

```
Advanced Params:
   Adv Params:
         Current Value: Disabled
         Default Value: Enabled
         Description: Nvm Dcbx Cfg
         Name: NvmDcbxCFg
         Param Type: String-Enum
         String Max Length: NA
         Supported Values: Disabled,Enabled
   Message: Success
   Reboot Required: true
```

The new settings will be active after reboot.

The current set of NIC advanced parameters for the 41000/45000 Series Adapters are read-only; for the 3400/8400 Series Adapters, they are read/write.

Adapter FCoE Namespace Commands and Sub-namespaces

To view FCoE namespace and sub-namespace commands, issue the following:

```
~# esxcli qlfastlinq qcc fcoe
```

Syntax

```
esxcli qlfastlinq qcc fcoe {cmd} [cmd options]
```

Namespaces

Available namespaces include the following:

Namespace	Description
`lun`	QCC CLI for FCoE LUNs
`target`	QCC CLI for FCoE targets
`vnport`	QCC CLI for FCoE virtual N_Port (VN_Port) information
`info`	QCC CLI for FCoE information

Available Commands

`list`
Lists identifiers for the list of Marvell Adapter FCoE ports discovered on the host. See List FCoE Ports.

List FCoE Ports

To view a list of all supported 3400/8400 Series and 41000/45000 Series FCoE ports, issue the `fcoe list` command.

Example

The following example shows the `fcoe list` command on a local host:

```
~# esxcli qlfastlinq qcc fcoe list
```

The following example shows the `fcoe list` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc fcoe
list
```

The output for both commands is:

```
FCoE
--------------------------
FCoE_10-00-00-0E-1E-C4-C5-EC
FCoE_10-00-00-0E-1E-C4-C5-ED
```

Adapter FCoE Info Namespace Commands and Sub-namespaces

```
~# esxcli qlfastlinq qcc fcoe info
```

Syntax

```
esxcli qlfastlinq qcc fcoe info {cmd} [cmd options]
```

Available Commands

`get`
Shows information about the specified Marvell Adapter FCoE function. See Get FCoE Information.

Get FCoE Information

To view FCoE information for the specified FCoE device, issue the following command:

```
~# esxcli qlfastlinq qcc fcoe info get
```

Syntax

```
esxcli qlfastlinq qcc fcoe info get [cmd options]
```

Keywords

`-i|--identifier <str>` = FCoE identifier (required). Obtain the identifier from the `fcoe list` command output.

Example

The following example shows the `fcoe info get` command on a local host:

```
~# esxcli qlfastlinq qcc fcoe info get -i
FCoE_10-00-00-0E-1E-C4-C5-ED
```

The following example shows the `fcoe info get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc fcoe
info get -i FCoE_10-00-00-0E-1E-C4-C5-ED
```

The output for both commands is:

```
FCoE Information:
   FCOEInfo:
```

```
            Driver Date: Sep 08, 2015
            Driver Name: qedf
            Driver Version: 1.2.8.8.18130
            FCFSelection: First Received
            Firmware Version: 08.18.14.00
            MACAddress: 00:0E:1E:C4:C5:ED
            MTU: 9000
            VNPorts: VNPort_20-00-00-0E-1E-C4-C5-ED
            WWNN: 10-00-00-0E-1E-C4-C5-ED
            WWPN: 20-00-00-0E-1E-C4-C5-ED
      Message: Success
```

Adapter FCoE VN_Port Namespace Commands and Sub-namespaces

```
~# esxcli qlfastlinq qcc fcoe vnport
```

Syntax

```
esxcli qlfastlinq qcc fcoe vnport {cmd} [cmd options]
```

Namespaces

Available namespaces include the following:

Namespace	Description
`info`	QCC CLI for FCoE VN_Port information
`stats`	QCC CLI for FCoE VN_Port statistics

Available Commands

`list`
Lists identifiers for the Marvell Adapter FCoE VN_Ports discovered on the host. See List VN_Ports.

List VN_Ports

The `fcoe vnport list` command lists all of the supported 3400/8400 Series and 41000/45000 Series FCoE VN_Ports.

Example

The following example shows the `fcoe vnport list` command on a local host:

```
~# esxcli qlfastlinq qcc fcoe vnport list
```

The following example shows the `fcoe vnport list` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc fcoe
vnport list
```

The output for both commands is:

```
VNPorts
------------------------------
VNPort_20-00-00-0E-1E-C4-C5-EC
VNPort_20-00-00-0E-1E-C4-C5-ED
```

Adapter FCoE VN_Port Info Namespace Commands and Sub-namespaces

```
~# esxcli qlfastlinq qcc fcoe vnport info
```

Syntax

```
esxcli qlfastlinq qcc fcoe vnport info {cmd} [cmd options]
```

Available Commands

`get`
Shows information about the specified Marvell Adapter FCoE VN_Port. See Get FCoE VN_Port Info.

Get FCoE VN_Port Info

To view FCoE VN_Port information of the specified FCoE VN_Port device, issue the following command:

```
~# esxcli qlfastlinq qcc fcoe vnport info get
```

Syntax

```
esxcli qlfastlinq qcc fcoe vnport info get [cmd options]
```

Keywords

`-i|--identifier <str>` = VN_Port identifier (required). Obtain the identifier from the `fcoe vnport list` command output.

Example

The following example shows the `fcoe vnport info get` command on a local host:

```
~# esxcli qlfastlinq qcc fcoe vnport info get -i
VNPort_20-00-00-0E-1E-C4-C5-ED
```

The following example shows the `fcoe vnport info get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc fcoe
vnport info get -i VNPort_20-00-00-0E-1E-C4-C5-ED
```

The output for both commands is:

```
VNPort Information:
   Message: Success
   VNPort Info:
         FCFMACAddress: 28:34:A2:E6:A2:30
         Fabric Name: 20-01-00-05-9B-23-D8-01
         Fabric Priority: 128
         Link Keep Alive Interval: 1669389911
         Parent WWPort Name: 20-00-00-0E-1E-C4-C5-ED
         Port FCID: 9634208
         Port State: Up
         Port Type: NPort
         Switch Name: 20-01-8C-60-4F-2D-3C-01
         Targets:
VNPort_20-00-00-0E-1E-C4-C5-ED_Target_50-0A-09-81-80-0F-06-39
         WWNode Name: 10-00-00-0E-1E-C4-C5-ED
         WWPort Name: 20-00-00-0E-1E-C4-C5-ED
```

View FCoE VN_Port Statistics

```
~# esxcli qlfastlinq qcc fcoe vnport stats
```

Syntax

```
esxcli qlfastlinq qcc fcoe vnport stats {cmd} [cmd options]
```

Available Commands

`get`
Shows statistics for the specified Marvell Adapter FCoE VN_Port. See Get FCoE VN_Port Statistics.

Get FCoE VN_Port Statistics

To view statistics for the specified FCoE VN_Port device, issue the following command:

```
~# esxcli qlfastlinq qcc fcoe vnport stats get
```

Syntax

```
esxcli qlfastlinq qcc fcoe vnport stats get [cmd options]
```

Keywords

`-i|--identifier <str>` = VN_Port identifier (required). Obtain the identifier from the `fcoe vnport list` command output.

Example

The following example shows the `fcoe vnport stats get` command on a local host:

```
~# esxcli qlfastlinq qcc fcoe vnport stats get -i
VNPort_20-00-00-0E-1E-C4-C5-ED
```

The following example shows the `fcoe vnport stats get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc fcoe
vnport stats get -i VNPort_20-00-00-0E-1E-C4-C5-ED
```

The output for both commands is:

```
VNPort Statistics:
   Message: Success
   VNPort Stats:
         Fcoe Dropped Frames: 0
         Fcoe Received Bytes: 440591
         Fcoe Received Frames: 16243
         Fcoe Transmitted Bytes: 224075
         Fcoe Transmitted Frames: 8963
         Wrong Fcoe Crc Packets: 0
```

Adapter FCoE Target Namespace Commands and Sub-namespaces

```
~# esxcli qlfastlinq qcc fcoe target
```

Syntax

```
esxcli qlfastlinq qcc fcoe target {cmd} [cmd options]
```

Namespaces

Available namespaces include the following:

Namespace	Description
`info`	QCC CLI for FCoE target information

Available Commands

`list`
Lists identifiers for the Marvell Adapter FCoE targets discovered on the host. See List FCoE Targets.

List FCoE Targets

The `fcoe target list` command lists all of the detected FCoE targets on 3400/8400 Series and 41000/45000 Series Adapters.

Example

The following example shows the `fcoe target list` command on a local host:

```
~# esxcli qlfastlinq qcc fcoe target list
```

The following example shows the `fcoe target list` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc fcoe
target list
```

The output for both commands is:

```
Targets
-------------------------------------------------------------
VNPort_20-00-00-0E-1E-C4-C5-ED_Target_50-0A-09-81-80-0F-06-39
```

Adapter FCoE Target Information Namespace Commands and Sub-namespaces

```
~# esxcli qlfastlinq qcc fcoe target info
```

Syntax

```
esxcli qlfastlinq qcc fcoe target info {cmd} [cmd options]
```

Available Commands

`get`
Shows information about the specified Marvell Adapter FCoE target. See Get FCoE Target Information.

Get FCoE Target Information

To view FCoE target information for the specified Marvell Adapter FCoE target device, issue the following command:

```
~# esxcli qlfastlinq qcc fcoe target info get
```

Syntax

```
esxcli qlfastlinq qcc fcoe target info get [cmd options]
```

Keywords

`-i|--identifier <str>` = Target identifier (required). Obtain the identifier from the `fcoe target list` command output.

Example

The following example shows the `fcoe target info get` command on a local host:

```
~# esxcli qlfastlinq qcc fcoe target info get -i
VNPort_20-00-00-0E-1E-C4-C5-ED_Target_50-0A-09-81-80-0F-06-39
```

The following example shows the `fcoe target info get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc fcoe
target info get -i
VNPort_20-00-00-0E-1E-C4-C5-ED_Target_50-0A-09-81-80-0F-06-39
```

The output for both commands is:

```
Target Information:
   Message: Success
   Target Info:
         Lun Count: 1
         Luns:
VNPort_20-00-00-0E-1E-C4-C5-ED_Target_50-0A-09-81-80-0F-06-39_LUN_0
         Port State: Up
         Scsi Target Number: 0
         Target Node Name: 50-0A-09-80-80-0F-06-39
         Target Port ID: 00-E6-01-20
         Target Port Name: 50-0A-09-81-80-0F-06-39
```

Adapter FCoE LUN Namespace Commands and Sub-namespaces

```
~# esxcli qlfastlinq qcc fcoe lun
```

Syntax

```
esxcli qlfastlinq qcc fcoe lun {cmd} [cmd options]
```

Namespaces

Available namespaces include the following:

Namespace	Description
`info`	QCC CLI for FCoE LUN information

Available Commands

`list`
Lists identifiers for the Marvell Adapter FCoE LUNs discovered on the host. See List FCoE LUNs.

List FCoE LUNs

The `fcoe lun list` command lists all FCoE LUNs detected on 3400/8400 Series and 41000/45000 Series Adapters.

Example

The following example shows the `fcoe lun list` command on a local host:

```
~# esxcli qlfastlinq qcc fcoe lun list
```

The following example shows the `fcoe lun list` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc fcoe
lun list
```

The output for both commands is:

```
Luns
------------------------------------------------------------------
VNPort_20-00-00-0E-1E-C4-C5-ED_Target_50-0A-09-81-80-0F-06-39_LUN_0
```

Adapter FCoE LUN Info Namespace Commands and Sub-namespaces

```
~# esxcli qlfastlinq qcc fcoe lun info
```

Syntax

```
esxcli qlfastlinq qcc fcoe lun info {cmd} [cmd options]
```

Available Commands

`get`
Shows information about the specified Marvell Adapter FCoE LUN. See Get FCoE LUN Information.

Get FCoE LUN Information

To view FCoE LUN information for the specified FCoE LUN device, issue the following command:

```
~# esxcli qlfastlinq qcc fcoe lun info get
```

Syntax

```
esxcli qlfastlinq qcc fcoe lun info get [cmd options]
```

Keywords

`-i|--identifier <str>` = LUN identifier (required). Obtain the identifier from the `fcoe lun list` command output.

Example

The following example shows the `fcoe lun info get` command on a local host:

```
~# esxcli qlfastlinq qcc fcoe lun info get -i
VNPort_20-00-00-0E-1E-C4-C5-ED_Target_50-0A-09-81-80-0F-06-39_LUN_0
```

The following example shows the `fcoe lun info get` command on a remote host:

```
password qlfastlinq qcc fcoe lun info get -i
VNPort_20-00-00-0E-1E-C4-C5-ED_Target_50-0A-09-81-80-0F-06-39_LUN_0
```

The output for both commands is:

```
Lun Information:
   Lun Info:
         Capacity In MB: 5242
         Device Type: Disk
         Product ID: LUN
         Product Rev Level: 811a
         Scsi Unit Number: 0
         Vendor ID: NETAPP
   Message: Success
```

Adapter iSCSI Namespace Commands and Sub-namespaces

To view iSCSI namespace and sub-namespace commands, issue the following:

```
~# esxcli qlfastlinq qcc iscsi
```

Syntax

```
esxcli qlfastlinq qcc iscsi {cmd} [cmd options]
```

Namespaces

Available namespaces include the following:

Namespace	Description
`portal`	QCC CLI for iSCSI portals
`target`	QCC CLI for iSCSI targets
`info`	QCC CLI for iSCSI target information

Available Commands

`list`
Lists identifiers for the list of Marvell Adapter iSCSI ports discovered on the host. See List iSCSI Ports.

List iSCSI Ports

The `iscsi list` command lists all supported 3400/8400 Series and 41000/45000 Series iSCSI ports.

Example

The following example shows the `iscsi list` command on a local host:

```
~# esxcli qlfastlinq qcc iscsi list
```

The following example shows the `iscsi list` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc iscsi
list
```

The output for both commands is:

```
iSCSI
----------------------
iSCSI_00:0E:1E:C4:B2:D9
iSCSI_00:0E:1E:C4:B2:DA
```

Adapter iSCSI Info Namespace Commands and Sub-namespaces

```
~# esxcli qlfastlinq qcc iscsi info
```

Syntax

```
esxcli qlfastlinq qcc iscsi info {cmd} [cmd options]
```

Available Commands

`get`
Shows information for the Marvell Adapter iSCSI port discovered on the host. See Get iSCSI Info.

Get iSCSI Info

To view iSCSI information for the specified Marvell Adapter iSCSI device port discovered on the host, issue the following command:

```
~# esxcli qlfastlinq qcc iscsi info get
```

Syntax

```
esxcli qlfastlinq qcc iscsi info get [cmd options]
```

Keywords

`-i|--identifier <str>` = iSCSI identifier (required). Obtain the identifier from the `iscsi list` command output.

Example

The following example shows the `iscsi info get` command on a local host:

```
~# esxcli qlfastlinq qcc iscsi info get -i iSCSI_00:0E:1E:C4:B2:DA
```

The following example shows the `iscsi info get` command on a remote host:

```
password qlfastlinq qcc iscsi info get -i  iSCSI_00:0E:1E:C4:B2:DA
```

The output for both commands is:

```
iSCSI Information:
   ISCSIInfo:
         Data Digest: Disabled
         Data Pdu In Order: Enabled
         Data Sequence In Order: Enabled
         Device Id: 0x165e
         Driver Name: qedil
         Driver Version: 1.0.8.2.0818.1300
         Error Recovery Level: 0
         Firmware Version: mfw 8.15.0.0 storm 8.18.14.0
         First Data Burst Length: 65536
         Header Digest: Disabled
         Immediate Data: Enabled
         Initial R2T: Disabled
         MACAddress: 00:0E:1E:C4:B2:DA
```

```
            MTU: 1500
            Max Outstanding R2T: 1
            Portals:
            IPv4Address: 192.168.100.124
            IPv4Port Number: 0
            IPv6Address: C0:A8:64:7C:00:00:00:00:00:00:00:00:00:00:00:00
            IPv6Port Number: 0
            Portal Number: 0
            Sessions:
            Session: 192.168.100.124 to 192.168.100.5
            Session Number: 0

            Session: 192.168.100.124 to 192.168.100.5
            Session Number: 1

            Session: 192.168.100.124 to 192.168.100.5
            Session Number: 2

            Session: 192.168.100.124 to 192.168.100.5
            Session Number: 3
      Message: Success
```

Adapter iSCSI Portal Namespace Commands and Sub-namespaces

```
~# esxcli qlfastlinq qcc iscsi portal
```

Syntax

```
esxcli qlfastlinq qcc iscsi portal {cmd} [cmd options]
```

Namespaces

Available namespaces include the following:

Namespace	Description
`info`	QCC CLI for iSCSI portal information

Available Commands

`list`
Lists portal identifiers for the Marvell Adapter iSCSI ports discovered on the host. See List iSCSI Portals.

`info`
Lists portal identifiers for the Marvell Adapter iSCSI ports discovered on the host. See List iSCSI Portals.

List iSCSI Portals

The `iscsi portal list` command lists all of the supported 3400/8400 Series and 41000/45000 Series iSCSI portals.

Example

The following example shows the `iscsi portal list` command on a local host:

```
~# esxcli qlfastlinq qcc iscsi portal list
```

The following example shows the `iscsi portal list` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc iscsi portal list
```

The output for both commands is:

```
Portals
---------------------
Portal_192.168.100.124
```

iSCSI Portal Info

```
~# esxcli qlfastlinq qcc iscsi portal info
```

Syntax

```
esxcli qlfastlinq qcc iscsi portal info {cmd} [cmd options]
```

Available Commands

`get`
Gets information for the Marvell Adapter iSCSI portal discovered on the host. See Get iSCSI Portal Info.

Get iSCSI Portal Info

To view iSCSI portal information for the specified Marvell Adapter iSCSI portal discovered on the host.

```
~# esxcli qlfastlinq qcc iscsi portal info get
```

Syntax

```
esxcli qlfastlinq qcc iscsi portal info get [cmd options]
```

Keywords

`-i|--identifier <str>` = iSCSI identifier (required). Obtain the identifier from the `iscsi portal list` command output.

Example

The following example shows the `iscsi portal info get` command on a local host:

```
~# esxcli qlfastlinq qcc iscsi portal info get -i
Portal_192.168.100.124
```

The following example shows the `iscsi portal info get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc iscsi
portal info get -i  Portal_192.168.100.124
```

The output for both commands is:

```
iSCSI Portal Information:
   Message: Success
   Portal Info:
         Portal IPAddress: 192.168.100.124
         Sessions:
         Connection Id: 4280287241
         Data Pdu In Order: Enabled
         Data Sequence In Order: Enabled
         Error Recovery Level: 0
         Immediate Data: Enabled
         Initial R2T: Disabled
         Initiator Portal: 192.168.100.124
         Max Outstanding R2T: 1
         Session Number: 0
         Session State: Connected
         Session Unique Id: 73685775427816
         Target:
iqn.2001-05.com.equallogic:0-8a0906-3a14b7e04-b5e000e7ac253579-isns
         Target Portal: 192.168.100.5

         Connection Id: 4280287235
         Data Pdu In Order: Enabled
         Data Sequence In Order: Enabled
         Error Recovery Level: 0
         Immediate Data: Enabled
         Initial R2T: Disabled
         Initiator Portal: 192.168.100.124
         Max Outstanding R2T: 1
         Session Number: 1
```

```
         Session State: Connected
         Session Unique Id: 73685775429208
         Target:
iqn.2001-05.com.equallogic:0-8a0906-4084b7e04-59f6c0a42cd5399c-
parag-broadcom
         Target Portal: 192.168.100.5

         Connection Id: 4280287242
         Data Pdu In Order: Enabled
         Data Sequence In Order: Enabled
         Error Recovery Level: 0
         Immediate Data: Enabled
         Initial R2T: Disabled
         Initiator Portal: 192.168.100.124
         Max Outstanding R2T: 1
         Session Number: 2
         Session State: Connected
         Session Unique Id: 73685775430600
         Target:
iqn.2001-05.com.equallogic:0-8a0906-be94b7e04-aa76c0a440c55cc9-
krishna-40g
         Target Portal: 192.168.100.5

         Connection Id: 4280287237
         Data Pdu In Order: Enabled
         Data Sequence In Order: Enabled
         Error Recovery Level: 0
         Immediate Data: Enabled
         Initial R2T: Disabled
         Initiator Portal: 192.168.100.124
         Max Outstanding R2T: 1
         Session Number: 3
         Session State: Connected
         Session Unique Id: 73685775434776
         Target:
iqn.2001-05.com.equallogic:0-8a0906-e8f4b7e04-a5e0014e31f584ea-sas
-nanoe4
         Target Portal: 192.168.100.5
```

Adapter iSCSI Target Namespace Commands and Sub-namespaces

```
~# esxcli qlfastlinq qcc iscsi target
```

Syntax

```
esxcli qlfastlinq qcc iscsi target {cmd} [cmd options]
```

Namespaces

Available namespaces include the following:

Namespace	Description
`info`	QCC CLI for iSCSI target information

Available Commands

`list`
Lists target identifiers of iSCSI targets discovered through Marvell Adapters. See List iSCSI Targets.

List iSCSI Targets

The `iscsi target list` command lists all detected iSCSI targets on 3400/8400 Series and 41000/45000 Series Adapters.

Example

The following example shows the `iscsi target list` command on a local host:

```
~# esxcli qlfastlinq qcc iscsi target list
```

The following example shows the `iscsi target list` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc iscsi
target list
```

The output for both commands is:

```
Targets
-----------------------------------------------
Portal_192.168.100.124_Target_SID_73685775427816
Portal_192.168.100.124_Target_SID_73685775429208
Portal_192.168.100.124_Target_SID_73685775430600
Portal_192.168.100.124_Target_SID_73685775434776
```

Adapter iSCSI Target Info Namespace Commands and Sub-namespaces

```
~# esxcli qlfastlinq qcc iscsi target info
```

Syntax

```
esxcli qlfastlinq qcc iscsi target info {cmd} [cmd options]
```

Available Commands

`get`
Shows information for the Marvell Adapter iSCSI target discovered on the host. See Get iSCSI Target Info.

Get iSCSI Target Info

To view iSCSI target information for the specified Marvell Adapter iSCSI target device discovered on the host, issue the following command:

```
~# esxcli qlfastlinq qcc iscsi target info get
```

Syntax

```
esxcli qlfastlinq qcc iscsi target info get [cmd options]
```

Keywords

`-i|--identifier <str>` = Target identifier (required). Obtain the identifier from the `iscsi target list` command output.

Example

The following example shows the `iscsi target info get` command on a local host:

```
~# esxcli qlfastlinq qcc iscsi target info get -i
Portal_192.168.100.124_Target_SID_73685775427816
```

The following example shows the `iscsi target info get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc iscsi
target info get -i
Portal_192.168.100.124_Target_SID_73685775427816
```

The output for both commands is:

```
iSCSI Target Information:
   Message: Success
   Target Info:
         IPv4Address: 192.168.100.5
         IPv6Address: 00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
         Lun Count: 1
```

```
            MACAddress: 00:09:8A:04:7E:4C
            Scsi Target Number: 0
            Sessions:
            Connection Id: 4280287241
            Data Pdu In Order: Enabled
            Data Sequence In Order: Enabled
            Error Recovery Level: 0
            Immediate Data: Enabled
            Initial R2T: Disabled
            Initiator Portal: 192.168.100.124
            Max Outstanding R2T: 1
            Session Number: 0
            Session State: Connected
            Session Unique Id: 73685775427816
            Target:
iqn.2001-05.com.equallogic:0-8a0906-3a14b7e04-b5e000e7ac253579-isns
            Target Portal: 192.168.100.5
            Target Iqn Name:
iqn.2001-05.com.equallogic:0-8a0906-3a14b7e04-b5e000e7ac253579-isns
```

Adapter iSCSI LUN Namespace Commands and Sub-namespaces

```
~# esxcli qlfastlinq qcc iscsi lun
```

Syntax

```
esxcli qlfastlinq qcc iscsi lun {cmd} [cmd options]
```

Namespaces

Available namespaces include the following:

Namespace	Description
`info`	QCC CLI for iSCSI LUN information

Available Commands

`list`

Lists identifiers for Marvell Adapter iSCSI LUNs on 3400/8400 Series and 41000/45000 Series adapters discovered on the host. See List iSCSI LUNs.

List iSCSI LUNs

The `iscsi lun list` command lists all detected iSCSI LUNs.

Example

The following example shows the `iscsi lun list` command on a local host:

```
~# esxcli qlfastlinq qcc iscsi lun list
```

The following example shows the `iscsi lun list` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc iscsi
lun list
```

The output for both commands is:

```
Luns
-------------------------------------------------------------------
Portal_192.168.100.124_Target_SID_73683392473320_LUN_0
Portal_192.168.100.124_Target_SID_73683392474712_LUN_0
Portal_192.168.100.124_Target_SID_73683392476104_LUN_0
Portal_192.168.100.124_Target_SID_73683392480280_LUN_0
```

Adapter iSCSI LUN Info Namespace Commands and Sub-namespaces

```
~# esxcli qlfastlinq qcc iscsi lun info
```

Syntax

```
esxcli qlfastlinq qcc iscsi lun info {cmd} [cmd options]
```

Available Commands

`get`
Shows information about the specified Marvell Adapter iSCSI LUN. See Get iSCSI LUN Info.

Get iSCSI LUN Info

To view iSCSI LUN information for the specified Marvell Adapter iSCSI LUN device, issue the following command:

```
~# esxcli qlfastlinq qcc iscsi lun info get
```

Syntax

```
esxcli qlfastlinq qcc iscsi lun info get [cmd options]
```

Keywords

`-i|--identifier <str>` = LUN identifier (required). Obtain the identifier from the `iscsi lun list` command output.

Example

The following example shows the `iscsi lun info get` command on a local host:

```
~# esxcli qlfastlinq qcc iscsi lun info get -i
Portal_192.168.100.124_Target_SID_73683392474712_LUN_0
```

The following example shows the `iscsi lun info get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfastlinq qcc iscsi
lun info get -i
Portal_192.168.100.124_Target_SID_73683392474712_LUN_0
```

The output for both commands is:

```
Lun Information:
   Lun Info:
         Capacity In MB: 5253
         Device Type: Disk
         Product ID: 100E-00
         Product Rev Level: 4.3
         Scsi Unit Number: 0
         Vendor ID: EQLOGIC
   Message: Success
```

4 Fibre Channel CLI Command Extensions

This chapter describes the functions of the various software components added to extend the capabilities of the Marvell ESXCLI VMware Plug-in software for the Marvell QLogic Fibre Channel Adapters.

Sections provided in this chapter include the following:

See Figure 3-1 for an illustration of the ESXCLI extension module flow chart. See "Extending ESXCLI Commands" on page 7 for a high level description of how to extend ESXCLI commands.

In this chapter:

- *SAN congestion management* (SCM) is a common noun, and describes a standards-based Fibre Channel technology.
- *Universal SAN Congestion Mitigation* (USCM) is Marvell's IP, and describes Marvell's SCM feature.

Extension Command Format

The ESXCLI commands are organized as a hierarchy of *namespaces*. If you are a VMware partner, and you are creating your own ESXCLI commands, organize the commands in a consistent, logical manner to improve usability of your own commands and of ESXCLI as a whole.

Usage Guidelines

If each partner adheres to the following guidelines, each partner can have full control of the organization of the namespaces and commands.

- **Single root namespace.** Each partner namespace has a single root namespace, which is the name of the partner, in lowercase letters. For example, if you belong to the Acme organization, the root namespace should be `acme`, and all namespaces and commands that you add to ESXCLI will be under `esxcli acme`.
- **Organized hierarchy.** Organize the commands in the namespace and use additional namespaces as needed. In most cases, your commands apply to a single product or product type. For example, if you are shipping commands to support the widget product, place the commands under `esxcli acme widget`. If you are shipping a second product, you might create a second namespace.
- **Commands are actions.** Each command should operate on a logical object. For example, a command `esxcli acme widget stats get` follows the convention, but `esxcli acme widget getstats` does not. Look at VMware ESXCLI commands for examples.
- **Consistent XML format.** Make the XML format consistent with other ESXCLI commands. For example:

```
$ esxcli --formatter=xml system time get
<?xml version="1.0" encoding="utf-8"?>
<output xmlns="http://www.vmware.com/Products/ESX/5.0/esxcli">
<root>
<string>2013-09-26T00:08:07Z</string>
</root>
</output>
```

Syntax

ESXCLI commands in this chapter have the following syntax:

```
esxcli [dispatcher options] <namespace1 [namespace2] ...>
<command> [input parameters]
```

- **Dispatcher options.** Predefined options for connection information such as target host, user name, and so on. Extension developers have no control over dispatcher options and do not need to be concerned about them.
- **Namespace.** Organizational structure that groups a quantity of ESXCLI objects. One namespace is required; additional namespaces are possible.

  Examples of namespaces include:

  - `nmp`
    VMware native multipathing commands.
  - `swiscsi`
    Commands in the software iSCSI namespace.
  - `corestorage`
    VMware core storage commands.
- **Command.** The command reports on or modifies the state on the system. Commands are the leaf nodes in the ESXCLI syntax. Commands are the verbs that act on the objects defined by the namespaces.
- **Input parameters.** Multiple input parameters are supported.

All ESXCLI commands in this section have examples of command usage with both a local host and a remote host. The remote host in the examples has the following credentials:

- IP address: 172.27.9.144
- Password: password
- User name: root

Format

For a local host, command syntax is as follows:

```
esxcli qlfc qcc <namespace> <command> <options>
```

For a remote host, the command syntax is as follows:

```
esxcli --server <ip_address> --username <username> --password
<password> qlfc qcc namespace [command] [options]
```

where `<ip_address>`, `<username>`, and `<password>` are the IP address and credentials for the remote host. The keywords `--server`, `--username`, and `--password` can be abbreviated `-s`, `-u`, and `-p` respectively.

To obtain detailed information for a specific command, type the command without arguments or with the `--help` option.

All ESXCLI commands in this section have examples of command usage with both a local host and a remote host. The remote host in the examples has the following credentials:

- IP address: 172.27.9.144
- Password: password
- User name: root

NOTE

For remote operation, the VMware vSphere CLI software must be installed on the local workstation. For more information about obtaining, installing, and operating the VMware software, see the VMware documentation for your system.

Extension Module Functional Specification

`qfcesxcli` is the standard command line executable module invoked by the Fibre Channel (FC) ESXCLI VMware Plug-in plug-in extension wrapper. The `qfcesxcli` command line executable module supports multiple sets of input arguments for different commands and produces XML output for the FC ESXCLI VMware Plug-in extension wrapper to consume.

The non-interactive commands that are supported by `qfcesxcli` must be published to the ESXCLI plug-in extension wrapper using an extension XML file that adheres to the metadata specification. Together, the `qfcesxcli` and the extension XML file enable users to view and execute commands through ESXCLI (both locally as well as remotely). All of the commands are published as part of the `qlfc.qcc` sub-namespace (ESXCLI namespace).

QCC Namespace-level Commands and Sub-namespaces

At the top-most level—the Marvell QConvergeConsole® (QCC) namespace level—the following commands and sub-namespaces are exposed:

```
~ # esxcli qlfc qcc
```

Syntax

```
esxcli qlfc qcc {cmd} [cmd options]
```

Namespaces

Available namespaces include the following:

Namespace	Description
`adapter`	QCC CLI for operations on adapters
`lun`	QCC CLI for operations on LUNs
`port`	QCC CLI for operations on port functions
`target`	QCC CLI for operations on target functions

Adapter Namespace Commands and Sub-namespaces

```
~# esxcli qlfc qcc adapter
```

Syntax

```
esxcli qlfc qcc adapter {cmd} [cmd options]
```

Namespaces

Available namespaces include the following:

Namespace	Description
`info`	QCC CLI for adapter information
`personality`	QCC CLI for personality information
`flashinfo`	QCC CLI for flash information

Available Commands

`list`
Lists identifiers for the Marvell Adapters discovered on the host. See List Adapters.

`update`
Updates the flash on a Marvell Adapter. See Update Adapters.

List Adapters

To list all supported Fibre Channel Adapters, issue the `list` command as follows on the local host:

```
~ # esxcli qlfc qcc adapter list
```

The following example shows the `list` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc adapter list
```

The output for both commands is:

```
Adapters
-------------------------
Adapter_QLE2772_AFD1923Y07491
Adapter_QLE2770_AFD1924Y07544
```

The output contains model name and serial number as part of the adapter identifier.

Update Adapters

To update the flash on a Marvell Adapter, issue the `adapter update` command as follows:

```
~ # esxcli qlfc qcc adapter update
```

Syntax

```
esxcli qlfc qcc adapter update [cmd options]
```

Keywords

`-f|--flashfile <str>` = Flash file (required)

`-i|--identifier <str>` = Adapter identifier (required). Obtain the identifier from the adapter `list` output.

`-m|--image-type <str>` = Image type (required)

Parameter	Value (strings are case-sensitive)
`identifier`	Adapter identifier from the `adapter list` command
`flashfile`	Flash file name with complete path
`image-type`	MBI, BrdCfg, PEPCfg, FWPreload

Example

The following example shows the `adapter update` command on a local host:

```
~ # esxcli qlfc qcc adapter update -i
Adapter_QLE2772_AFD1923Y07491 -f /scratch/mh010205.bin -m MBI
```

The flash file must be on the remote host before issuing the `adapter update` command as described in the following steps.

1. Determine the name of the volume on the remote host in which to store the flash file. You will use the volume name and VMFS UUID in subsequent steps.

```
# esxcli -s 172.27.9.144 -u root -p password storage vmfs extent list
Volume Name    VMFS UUID                            Extent Number    Device Name
-------------- ------------------------------------ -------------    -----------
datastore1 (3) 4d435b11-2d3d0b17-a566-000af73e458c                   0
```

2. Create a temporary directory on the remote host in which to store the flash file using the name of the volume that you obtained in Step 1. The following command creates a directory called `tmp` on the volume `datastore1 (3)`. You will specify this directory when you update the flash in Step 4.

```
# vifs.pl --server 172.27.9.144 --username root --password
password --mkdir "[datastore1 (3)] tmp"
```

3. Upload the flash file to the remote host.

```
# vifs.pl --server 172.27.9.144 --username root --password
password --put mh010205.bin "[datastore1 (3)]
tmp/mh010205.bin"
```

4. Update the adapter flash on the remote host. The flash file path is indicated in `/vmfs/volumes/<vmfs_uuid>/tmp/mh010205.bin`, where the value for `<vmfs_uuid>` comes from Step 1.

The following example shows the `adapter update` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc adapter
update -i Adapter_QLE2772_AFD1923Y07491 -f /vmfs/volumes/
4d435b11-2d3d0b17-a566-000af73e458c/tmp/mh010205.bin -m MBI
```

The output for both commands is:

```
Message: Success
Reboot Required: true
```

Adapter Information

To obtain information about a specific Marvell Adapter, issue the `info` command as follows:

```
~ # esxcli qlfc qcc adapter info
```

Syntax

```
esxcli qlfc qcc adapter info {cmd} [cmd options]
```

Available Commands

`get`
Shows information about the adapter. See Get Adapter Information.

Get Adapter Information

To obtain information about a Marvell Adapter, issue the `get` command as follows:

```
~# esxcli qlfc qcc adapter info get
```

Syntax

```
esxcli qlfc qcc adapter info get [cmd options]
```

Keywords

`-i|--identifier <str>` = Adapter identifier (required). Obtain the identifier from the adapter `list` output.

Example

The following example shows the `adapter info get` command on a local host:

```
~# esxcli qlfc qcc adapter info get -i
Adapter_QLE2772_AFD1923Y07491
```

The following example shows the `adapter info get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc adapter info
get -i Adapter_QLE2772_AFD1923Y07491
```

The output for both commands is:

```
Adapter Information:
   General Parameters:
            Data:
      Manufacturer: QLogic Corporation
      Model: QLE2772
      Serial Number: AFD1923Y07491
      Vendor ID: 0x1077
      Device ID: 0x2281
      Subsystem Device ID: 0x2f3
      Subsystem Vendor ID: 0x1077
      PCI Bus ID: 0x37
      Chip Revision: 2
      Type: Fibre Channel Adapter
```

```
        Multiboot Version: 2.06.19
        Flash Firmware Version: 9.09.00
        Flash BIOS Version: 0.00
        Flash FCode Version: NA
        Flash EFI Version: 7.26
        Flash Preload Firmware Area Version: 4.01.15
        Flash FC Board Config Version: 3.07.02
        Flash MPI Firmware Version: 3.03.03
        Flash PEP Firmware Version: 3.01.36
        Flash PEP Board Config Version: 3.01.00
        Flash PEP SoftROM Version: 3.00.13
        Flash MPI SoftROM Version: 255.255.255
        Flash PCIE Serdes Version: 3.00.07

Message: Success
```

Adapter Personality

To display and set the adapter's current personality (FC or CNA), issue the `adapter personality` namespace command as follows:

```
~# esxcli qlfc qcc adapter personality
```

Syntax

```
esxcli qlfc qcc adapter personality {cmd} [cmd options]
```

Available Commands

`get`
Shows personality information for the specified Marvell Adapter. See Get Adapter Personality.

`set`
Sets the personality for the specified Marvell Adapter. See Set Adapter Personality.

Get Adapter Personality

NOTE

Personality commands are only supported on 2670/8300 Series Marvell Adapters.

To read the personality of a Marvell Adapter, issue the `adapter personality get` command as follows:

```
~# esxcli qlfc qcc adapter personality get
```

Syntax

```
esxcli qlfc qcc adapter personality get [cmd options]
```

Keywords

`-i|--identifier <str>` = Adapter identifier (required). Obtain the identifier from the adapter `list` output.

Example

The following example shows the `adapter personality get` command on a local host:

```
~# esxcli qlfc qcc adapter personality get -i
Adapter_QLE8362_MY58330JGY
```

The following example shows the `adapter personality get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc adapter
personality get -i Adapter_QLE8362_MY58330JGY
```

The output for both commands is:

```
Adapter Personality:
Message: Success
Personality: FC
```

Set Adapter Personality

NOTE

The personality `set` command is only supported on 2670/8300 Series Marvell Adapters.

To set the personality of a Marvell Adapter, issue the `adapter personality set` command as follows:

```
~# esxcli qlfc qcc adapter personality set
```

Syntax

```
esxcli qlfc qcc adapter personality set [cmd options]
```

Keywords

`-i|--identifier <str>` = Adapter identifier (required). Obtain the identifier from the adapter `list` output.

`-p|--personality <str>` = Adapter personality to be set (required)

Parameter	Value (strings are case-sensitive)
`identifier`	Adapter identifier from the `adapter list` command
`personality`	FC, CNA

Example

The following example shows the `adapter personality set` command on a local host:

```
~# esxcli qlfc qcc adapter personality set -i
Adapter_QLE8362_MY58330JGY
```

The following example shows the `adapter personality set` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc adapter
personality set -i Adapter_QLE8362_MY58330JGY -p CNA
```

The output for both commands is:

```
Adapter Personality:
Message: Success
Personality: CNA
Reboot Required: true
```

Adapter Flash Information

To obtain flash information about a specific Marvell Adapter, issue the `flashinfo` command as follows:

```
~ # esxcli qlfc qcc adapter flashinfo
```

Syntax

```
esxcli qlfc qcc adapter flashinfo {cmd} [cmd options]
```

Available Commands

`get`
Shows the flash information of the specified Marvell Adapter. See Get Adapter Flash Information.

Get Adapter Flash Information

To obtain flash information about a Marvell Adapter, issue the `get` command as follows:

```
~# esxcli qlfc qcc adapter flashinfo get
```

Syntax

```
esxcli qlfc qcc adapter flashinfo get [cmd options]
```

Keywords

`-i|--identifier <str>` = Adapter identifier (required). Obtain the identifier from the adapter `list` output.

Example

The following example shows the adapter `flashinfo get` command on a local host:

```
~# esxcli qlfc qcc adapter flashinfo get -i
Adapter_QLE2772_AFD1923Y07491
```

The following example shows the `adapter flashinfo get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc adapter
flashinfo get -i Adapter_QLE2772_AFD1923Y07491
```

The output for both commands is:

```
Adapter Flash Information:
   Flash Info:
         Data:
      MBI Version: 02.04.07
      MBI Build Date: 2020-11-17
      Flash Tool ID: FC PowerKit
      Flash Tool Version: 01.00.16.00
      Last Update Time: 2021-06-16 09:35:32

   Message: Success
```

Adapter Port Namespace Commands and Sub-namespaces

This section provides information on the following:


```
# esxcli qlfc qcc port
```

Syntax

```
esxcli qlfc qcc port {cmd} [cmd options]
```

Namespaces

Available namespaces include the following:

Namespace	Description
`info`	QCC CLI for port information
`stats`	QCC CLI for statistics information
`dmi`	QCC CLI for SFP DMI information
`vpd`	QCC CLI for VPD information
`hbaparams`	QCC CLI for HBA parameters configuration
`bbcr`	QCC CLI for buffer-to-buffer credit (BBCR) configuration
`fec`	QCC CLI for FEC information
`bootconfig`	QCC CLI for boot configuration
`bootconfigext`	QCC CLI for boot configuration
`fwdump`	QCC CLI for firmware dump information
`beacon`	QCC CLI for beacon configuration
`dportdiag`	QCC CLI for diagnostic port (DPort)
`ctdiag`	QCC CLI for common pass-thru (CT) diagnostics commands
`fcecho`	QCC CLI for FC Echo ELS diagnostic commands
`fcping`	QCC CLI for FC ping diagnostics commands
`rdp`	QCC CLI for FC port read diagnostics parameters information
`loopback`	QCC CLI for loopback diagnostics commands
`rwbuffer`	QCC CLI for FC read/write buffer diagnostics commands
`scmchk`	QCC CLI for FC port Universal SAN Congestion Mitigation (USCM) status
`scmstats`	QCC CLI for USCM statistics
`scmprofile`	QCC CLI for USCM profile

Available Commands

`list`
Lists identifiers for the Marvell Adapter ports discovered on the host.

`updatenvram`
Updates NVRAM data of Marvell Adapter ports discovered on the host. See “Adapter Port Update NVRAM” on page 149.

Adapter Port List

The `port list` command (shown issued from a local host) lists all of the supported 2770 Series ports, as follows:

```
~# esxcli qlfc qcc port list
```

The following example shows the `port list` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port list
```

The output for both commands is:

```
Ports
-------------------------
FC_51:40:2e:c0:12:3c:f4:a0
FC_51:40:2e:c0:12:3c:f4:a2
FC_51:40:2e:c0:01:c7:ed:18
FC_51:40:2e:c0:01:c7:ed:1a
```

Adapter Port Update NVRAM

The `port updatenvram` command updates the NVRAM on a Marvell Adapter port:

```
~ # esxcli qlfc qcc port updatenvram
```

Syntax

```
esxcli qlfc qcc port updatenvram [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

`-f|--nvramfile <str>` = NVRAM file (required)

Example

The following example shows the `port updatenvram` command on a local host:

```
~ # esxcli qlfc qcc port updatenvram -i FC_51:40:2e:c0:12:3c:f4:a0
-f /scratch/FCN.dat
```

The following example shows the `port updatenvram` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
updatenvram -i FC_51:40:2e:c0:12:3c:f4:a0 -f /scratch/FCN.dat
```

The output for both commands is:

```
Message: Success
Reboot Required: true
```

Adapter Port Information

The `port info` command shows information about the selected Marvell Adapter port:

~# esxcli qlfc qcc port info

Syntax

```
esxcli qlfc qcc port info {cmd} [cmd options]
```

Available Commands

`get`
Shows information about the specified Marvell Adapter port. See Get Port Information.

Get Port Information

The `port info get` command shows port-level information for the specified Marvell Adapter port.

~# esxcli qlfc qcc port info get

Syntax

```
esxcli qlfc qcc port info get [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port info get` command on a local host:

~# esxcli qlfc qcc port info get -i FC_51:40:2e:c0:12:3c:f4:a0

The following example shows the `port info get` command on a remote host:

 # esxcli -s 172.27.9.144 -u root -p password qlfc qcc port info get -i FC_51:40:2e:c0:12:3c:f4:a0

The output for both commands is:

```
Port Information:
   Message: Success
   Port Info:
```

```
  Data:
Model: QLE2772
Port Number: 1
PCI Bus Number: 0x37
PCI Device Number: 0x0
PCI Function Number: 0x0
PCI Bus Max Width: x8
PCI Bus Negotiated Width: x8
PCI Bus Max Speed: 16 Gtps
PCI Bus Negotiated Speed: 8 Gtps
Driver Version: 4.1.64
Host NQN: nqn.2014-08.org.nvmexpress:uuid:
          32324c58-6e35-3154-3943-505030303930
Host ID: 7286afd3b3ad45da83aa4f2cbdb3c10e
Node Name: 51:40:2e:c0:12:3c:f4:a1
Port Name: 51:40:2e:c0:12:3c:f4:a0
Principal Fabric WWN: 10:00:88:94:71:97:67:63
Adjacent Fabric WWN: 20:11:88:94:71:97:67:63
Maximum Speed: 32 Gbps
Supported Speeds: 8 16 32 Gbps
Port Interface Type: Physical Port
Protocol Type: FC
Port Type: Node Port
Port State: Online
Port Connection Mode: Point to Point
Port Speed: 32 Gbps
Port ID: 0a:11:00
Temperature In Celsius: 37
Flash Firmware Version: 9.09.00
Flash BIOS Version: 0.00
Flash FCode Version: NA
Flash EFI Version: 7.26
Flash Preload Firmware Area Version: 4.01.15
Flash FC Board Config Version: 3.07.02
Flash MPI Firmware Version: 3.03.03
Flash PEP Firmware Version: 3.01.36
Flash PEP Board Config Version: 3.01.00
Flash PEP SoftROM Version: 3.00.13
Flash MPI SoftROM Version: 255.255.255
Flash PCIE Serdes Version: 3.00.07
```

```
Running Firmware Version:  9.09.00
Running MPI Firmware Version: 3.03.03
Running PEP Firmware Version: 3.01.36
Config Lockdown: false
Firmware Update Lockdown: false
MPI Lockdown: false
```

Adapter Port Statistics

The `port stats` command shows statistics of the selected Marvell Adapter port:

`~# esxcli qlfc qcc port stats`

Syntax

`esxcli qlfc qcc port stats {cmd} [cmd options]`

Available Commands

`get`
Shows statistics of the specified Marvell Adapter port. See Get Port Statistics.

`reset`
Resets the statistics of the specified Marvell Adapter port. See Reset Port Statistics.

Get Port Statistics

The `port stats get` command shows port-level statistics for the specified port on the Marvell Adapter:

`~# esxcli qlfc qcc port stats get`

Syntax

`esxcli qlfc qcc port stats get [cmd options]`

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

`-m|--mode <str>` = Mode (required)

Parameter	Value (strings are case-sensitive)
`identifier`	Port identifier from the port `list` command
`mode`	Driver, Firmware

Example

The following example shows the `port stats get` command on a local host for Driver mode:

```
~# esxcli qlfc qcc port stats get -i FC_51:40:2e:c0:12:3c:f4:a0
-m Driver
```

The following example shows the `port stats get` command on a remote host for Driver mode:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port stats
get -i FC_51:40:2e:c0:12:3c:f4:a0 -m Driver
```

The output for both commands is:

```
FC Statistics:
   FCStats:
            Data:
         Controller Error Count: 0
         Device Error Count: 0
         IO Count: 1676
         Mega Bytes Count: 4
         LIP Reset Count: 0
         Interrupt Count: 0
   Message: Success
```

The following example shows the `port stats get` command on a local host for Firmware mode:

```
~# esxcli qlfc qcc port stats get -i FC_51:40:2e:c0:12:3c:f4:a0
-m Firmware
```

The following example shows the `port stats get` command on a remote host for Firmware mode:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port stats
get -i FC_51:40:2e:c0:12:3c:f4:a0 -m Firmware
```

The output for both commands is:

```
FC Statistics:
   FCStats:
         Data:
      Link Failure Count: 2
      Loss Of Sync Count: 4
      Loss Of Signal Count: 4
      Primitive Sequence Protocol Error Count: 0
      Invalid Transmission Word Count: 0
```

```
        Invalid CRC Count: 0

    Message: Success
```

Reset Port Statistics

The `port stats reset` command resets port-level statistics for the specified port on the Marvell Adapter.

```
~# esxcli qlfc qcc port stats reset
```

Syntax

```
esxcli qlfc qcc port stats reset [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

`-m|--mode <str>` = Mode (required)

Parameter	Value (strings are case-sensitive)
`identifier`	Port identifier from the port `list` command
`mode`	Firmware

Example

The following example shows the `port stats reset` command on a local host for Firmware mode:

```
~# esxcli qlfc qcc port stats reset -i FC_51:40:2e:c0:12:3c:f4:a0
-m Firmware
```

The following example shows the `port stats reset` command on a remote host for Firmware mode:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port stats
reset -i FC_51:40:2e:c0:12:3c:f4:a0 -m Firmware
```

The output for both commands is:

```
Reset FC Statistics:
   Message: Success
```

Adapter Port SFP DMI Information

The `port dmi` command shows SFP DMI information of the selected Marvell Adapter port:

~# esxcli qlfc qcc port dmi

Syntax

`esxcli qlfc qcc port dmi {cmd} [cmd options]`

Available Commands

`get`
Shows SFP DMI information of the specified Marvell Adapter port. See Get Port SFP DMI Information.

`getraw`
Shows raw SFP DMI information of the specified Marvell Adapter port. See Get Port Raw SFP DMI Information.

Get Port SFP DMI Information

The `port dmi get` command shows port-level SFP DMI Information for the specified Marvell Adapter port.

~# esxcli qlfc qcc port dmi get

Syntax

`esxcli qlfc qcc port dmi get [cmd options]`

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port dmi get` command on a local host:

~# esxcli qlfc qcc port dmi get -i FC_51:40:2e:c0:12:3c:f4:a0

The following example shows the `port dmi get` command on a remote host:

 # esxcli -s 172.27.9.144 -u root -p password qlfc qcc port dmi get -i FC_51:40:2e:c0:12:3c:f4:a0

The output for both commands is:

```
DMI Information:
   Dmi Info:
         Dmi Data:

------------------------------------------------------------------
```

```
Media Information
-----------------------------------------------------------------
                      Vendor: FINISAR CORP.
                   Connector: LC (Lucent Connector)
                  Media Type: 800-M5-SN-S
                 Part Number: FTLF8532P4BCV-QL
                       Speed: 3200 MBytes/Sec 1600 MBytes/Sec 800 MBytes/Sec
                    Revision: A
               Serial Number: PY12TQ5
                  Identifier: SFP/SFP+/SFP28 and later
  Extended Compliance Codes: Unspecified
            Rate Identifier: FC-PI-6 (32/16/8G Independent
                             Rx, Tx Rate_Select)
       QLogic SFP Installed: Yes
-----------------------------------------------------------------

                Temperature   Voltage     Tx Bias    Tx Power    Rx Power

                    (C)         (V)         (mA)        (mW)        (mW)

-------------   -------     -------    --------    --------      -------

        Value      36.97      3.32        7.88       0.7327        0.7414

       Status     Normal       Normal      Normal      Normal      Normal

   High Alarm      75.00         3.60       12.00      1.9953      1.9953

 High Warning      70.00         3.50       11.50      1.5849      1.5849

  Low Warning       0.00         3.10        2.00      0.1585      0.0158

    Low Alarm      -5.00         3.00        1.00      0.1259      0.0100

Optical Transceiver Digital Diagnostic Data:

Address A0
           Identifier: SFP/SFP+/SFP28 and later
       Ext Identifier: GBIC/SFP defined by serial ID only
            Connector: LC (Lucent Connector)
```

```
       Ethernet Speed:
           Compliance: 0x00 0x00 0x00
       FC Link Length: Short Distance (S)
  FC Transmitter Tech: Shortwave Laser w/o OFC (SN)
FC Transmission Media: Multi-mode 50m (M5)
             FC Speed: 3200 MBytes/Sec 1600 MBytes/Sec 800 MBytes/Sec
             Encoding: 64B66B
          BR, Nominal: 0xff
      Rate Identifier: FC-PI-6 (32/16/8G Independent Rx, Tx Rate_Select
    Length (9um) - km: 0x00
         Length (9um): 0x00
        Length (50um): 0x03
      Length (62.5um): 0x00
  Length
  (50um, OM4, Copper): 0x0a
  Length
  (50um, OM3, Copper): 0x07
         Vendor name: FINISAR CORP.
Extended Compliance
Codes                : Unspecified
           Vendor OUI: 0x00 0x90 0x65
            Vendor PN: FTLF8532P4BCV-QL
           Vendor rev: A
           Wavelength: 0x0352
              CC_BASE: 0x1f
Optional Transceiver
              Signals: -Linear Receiver Output Implemented: 0x0
                       -Power Level Declaration: 0x0
                       -Cooled Transceiver Declaration: 0x0
                       -Retimer or CDR indicator: 0x1
                       -Paging implemented indicator: 0x0
                       -High Power Level Declaration by bit 1: 0x0
                       -High Power Level Declaration by bits 1 and 5: 0x0
                       -Reserved: 0x0
                      -Signal Loss, as defined in SFP MSA: 0x1
                      -Signal Loss, inverted from SFP MSA: 0x0
                      -TX_FAULT signal implemented: 0x1
                      -TX_DISABLE implemented and disables serial o/p: 0x1
                      -RATE_SELECT implemented: 0x1
                      -Tunable transmitter technology: 0x0
```

```
                     -Receiver decision threshold implemented: 0x00
   Signaling Rate Max: 0x70
   Signaling Rate Min: 0x00
            Vendor SN: PY12TQ5
            Date code: 170709
 Diag Monitoring Type:-Address change required: 0x0
                     -Power Measurement: 0x1
                     -Externally Calibrated: 0x0
                     -Internally Calibrated: 0x1
                     -Digital diag monitoring: 0x1
                     -Legacy diagnostic: 0x0
     Enhanced Options:-Optional Soft Rate Select ctrl per SFF-8431: 0x1
                     -Optional Application Select ctrl per SFF-8079: 0x0
                     -Optional Soft RATE_SELECT ctrl and monitoring: 0x1
                     -Soft RX_LOS monitoring: 0x1
                     -Soft TX_FAULT monitoring: 0x1
                     -Soft TX_DISABLE ctrl and monitoring: 0x1
                     -Alarm/warning flags: 0x1
  SFF-8472 Compliance: Includes functionality described in Rev 12.4 of SFF-8472
              CC_EXT: 0x9b
      Vendor Specific: 0x51 0x4c 0x6f 0x67 0x69 0x63 0x00 0x00
                       0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
                       0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
                       0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
 SFP Firmware Version: 0x00 0x00
 MCU Firmware Version: 0x00 0x00
 DSP Firmware Version: 0x00 0x00 0x00

Address A2
      Temp High Alarm: 0x4b00
       Temp Low Alarm: 0xfb00
    Temp High Warning: 0x4600
     Temp Low Warning: 0x0000
   Voltage High Alarm: 0x8ca0
    Voltage Low Alarm: 0x7530
 Voltage High Warning: 0x88b8
  Voltage Low Warning: 0x7918
      Bias High Alarm: 0x1770
       Bias Low Alarm: 0x01f4
    Bias High Warning: 0x1676
```

```
     Bias Low Warning: 0x03e8
TX Signal Power High Alarm: 0x4df1
TX Signal Power Low Alarm: 0x04eb
TX Signal Power High Warning: 0x3de9
TX Signal Power Low Warning: 0x0631
RX Signal Power High Alarm: 0x4df1
RX Signal Power Low Alarm: 0x0064
RX Signal Power High Warning: 0x3de9
RX Signal Power Low Warning: 0x009e
            Rx_PWR(4): 0x00000000
            Rx_PWR(3): 0x00000000
            Rx_PWR(2): 0x00000000
            Rx_PWR(1): 0x3f800000
            Rx_PWR(0): 0x00000000
          Tx_I(Slope): 0x0100
         Tx_I(Offset): 0x0000
        Tx_PWR(Slope): 0x0100
       Tx_PWR(Offset): 0x0000
             T(Slope): 0x0100
            T(Offset): 0x0000
             V(Slope): 0x0100
            V(Offset): 0x0000
             Checksum: 0xd4
      Temperature MSB: 0x24
      Temperature LSB: 0xf8
              Vcc MSB: 0x81
              Vcc LSB: 0xb2
          TX Bias MSB: 0x0f
          TX Bias LSB: 0x66
         TX Power MSB: 0x1c
         TX Power LSB: 0x9f
         RX Power MSB: 0x1c
         RX Power LSB: 0xf6
         Reserved MSB: 0x00
         Reserved LSB: 0x00
         Reserved MSB: 0x00
         Reserved LSB: 0x00
  Status/Control Bits:-Data_Ready_Bar: 0x0
                      -LOS: 0x0
```

```
                        -TX Fault: 0x0
                        -Soft RX Rate Select: 0x0
                        -RX Rate Select State: 0x0
                        -Soft TX Disable: 0x0
                        -TX Disable State: 0x0
                        -EWRAP Control Bit: 0x0
                        -EWRAP FORWARD Control Bit: 0x0
                        -OWRAP Control Bit: 0x0
                        -OWRAP FORWARD Control Bit: 0x0
                        -TX Signal Power Low Alarm: 0x0
                        -TX Signal Power High Alarm: 0x0
                        -TX Bias Low Alarm: 0x0
                        -TX Bias High Alarm: 0x0
                        -Vcc Low Alarm: 0x0
                        -Vcc High Alarm: 0x0
                        -Temp Low Alarm: 0x0
                        -Temp High Alarm: 0x0
                        -RX Signal Power Low Alarm: 0x0
                        -RX Signal Power High Alarm: 0x0
                        -TX Signal Power Low Warning: 0x0
                        -TX Signal Power High Warning: 0x0
                        -TX Bias Low Warning: 0x0
                        -TX Bias High Warning: 0x0
                        -Vcc Low Warning: 0x0
                        -Vcc High Warning: 0x0
                        -Temp Low Warning: 0x0
                        -Temp High Warning: 0x0
                        -Reserved Warning: 0x0
                        -Reserved Warning: 0x0
                        -Reserved Warning: 0x0
                        -Reserved Warning: 0x0
                        -Reserved Warning: 0x0
                        -Reserved Warning: 0x0
                        -RX Signal Power Low Warning: 0x0
                        -RX Signal Power High Warning: 0x0
Extended Module Control: -Optional Power Level Select: 0x0
                        -Optional Power Level Operation State: 0x0
                        -Power Level 4 Enable: 0x0
                        -Soft RS(1) Select: 0x0
```

```
                                -Adaptive Input EQ Fail Flag: 0x0
                                -Reserved: 0x0
                                -Reserved: 0x0
                                -Reserved: 0x0
            Extended Status Bytes: -Optional Tx CDR Unlocked: 0x0
                                -64GFC Mode: 0x0
                                -PAM4 Mode Rx Configured: 0x0
                                -PAM4 Mode Tx Configured: 0x0
                                -Reserved: 0x0
                                -Reserved: 0x0
                                -Reserved: 0x0
                  Vendor Specific: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x01
         TP1 to TP4 EWRAP Control: -EWRAP Disable: 0x0
                                -EWRAP Enable: 0x0
         TP3 to TP2 OWRAP Control: -OWRAP Disable: 0x0
                                -OWRAP Enable: 0x0
 Electrical Output Tx Tap Pre3: 0x00
 Electrical Output Tx Tap Pre2: 0x00
 Electrical Output Tx Tap Pre1: 0x00
 Electrical Output Tx Tap Main: 0x00
Electrical Output Tx Tap Post1: 0x00
     DSP Status Timing Control: 0x00
                   DSP Control: 0x00
             FC 64G Mode Control: 0x00
                   LSN Control: -Train 64G: 0x0
                                -Train 32G: 0x0
                                -Fixed Speed Switch to 64G: 0x0
                                -Reserved: 0x0
                                -Reserved: 0x0
                                -Reserved: 0x0
                                -Client Rx Adaptation Reset: 0x0
                                -LSN Mode: 0x0
            CDR Firmware Version: 0x00 0x00 0x00
                  Laser Version: 0x00
                    TIA Version: 0x00
                    CDR Version: 0x00
                     HW Version: 0x00
                    MCU Version: 0x00
                PLP API Version: 0x00
```

```
            Debug MCU Version: 0x00

   Message: Success
```

Get Port Raw SFP DMI Information

The `port dmi getraw` command shows port-level raw SFP DMI information for the specified Marvell Adapter port:

~# esxcli qlfc qcc port dmi getraw

Syntax

`esxcli qlfc qcc port dmi getraw [cmd options]`

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port dmi getraw` command on a local host:

~# esxcli qlfc qcc port dmi getraw -i FC_51:40:2e:c0:12:3c:f4:a0

The following example shows the `port dmi getraw` command on a remote host:

 # esxcli -s 172.27.9.144 -u root -p password qlfc qcc port dmi getraw -i FC_51:40:2e:c0:12:3c:f4:a0

The output for both commands is:

```
DMI Information:
   Dmi Info:
         Dmi A0Data In Hex: 03 04 07 00 00 00 00 40 40 04 68 06 FF 0C 00 00
03 00 0A 07 46 49 4E 49 53 41 52 20 43 4F 52 50 2E 20 20 20 00 00 90 65 46 54
4C 46 38 35 33 32 50 34 42 43 56 2D 51 4C 41 20 20 20 03 52 00 1F 08 3A 70 00
50 59 31 32 54 51 35 20 20 20 20 20 20 20 20 20 31 37 30 37 30 39 20 20 68 FA
09 9B 51 4C 6F 67 69 63 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00
         Dmi A2Data In Hex: 4B 00 FB 00 46 00 00 00 8C A0 75 30 88 B8 79 18
17 70 01 F4 16 76 03 E8 4D F1 04 EB 3D E9 06 31 4D F1 00 64 3D E9 00 9E 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
3F 80 00 00 00 00 00 00 01 00 00 00 01 00 00 00 01 00 00 00 01 00 00 00 00 00
```

```
00 D4 24 62 81 51 0C C5 19 E3 12 18 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00
   Message: Success
```

Adapter Port VPD Information

The `port vpd` command shows VPD information of the selected Marvell Adapter port:

`~# esxcli qlfc qcc port vpd`

Syntax

`esxcli qlfc qcc port vpd {cmd} [cmd options]`

Available Commands

`get`
Shows VPD information of the specified Marvell Adapter port. See Get Port VPD.

Get Port VPD

The `port vpd get` command shows port-level VPD information for the specified Marvell Adapter port:

`~# esxcli qlfc qcc port vpd get`

Syntax

`esxcli qlfc qcc port vpd get [cmd options]`

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port vpd get` command on a local host:

`~# esxcli qlfc qcc port vpd get -i FC_51:40:2e:c0:12:3c:f4:a0`

The following example shows the `port vpd get` command on a remote host:

`# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port vpd get -i FC_51:40:2e:c0:12:3c:f4:a0`

The output for both commands is:

```
VPD Information:
```

```
Message: Success
VPDData:
      Tag: Engineering Date Code
      Value: 1-5917

      Tag: Manufacturing Id
      Value: MA2810401-34 03

      Tag: Product Identifier
      Value: QLogic 2x32Gb QLE2772 FC HBA

      Tag: Part Number
      Value: QLE2772

      Tag: Serial Number
      Value: AFD1923Y07491

      Tag: Miscellaneous Info
      Value: PW=15 Watts
```

Adapter Port HBA Parameters Configuration

The `port hbaparams` command configures the HBA parameters of the selected Marvell Adapter port:

`~# esxcli qlfc qcc port hbaparams`

Syntax

`esxcli qlfc qcc port hbaparams {cmd} [cmd options]`

Available Commands

`get`
Shows the HBA parameters information of the specified Marvell Adapter port. See Get Port HBA Parameters.

`setFile`
Sets the HBA parameters information of the specified Marvell Adapter port in a local cache file. See SetFile Port HBA Parameters.

`set`
Sets the HBA parameters information of the specified Marvell Adapter port from a local cache file to flash on the adapter. See Set Port HBA Parameters.

`restoreDefaults`
Sets the default values of the HBA parameters information of the specified Marvell Adapter port. See Restore Default Port HBA Parameters.

`save`
Saves HBA parameters information of the specified Marvell Adapter port to a user-provided file in text format. See Save Port HBA Parameters.

Get Port HBA Parameters

The `port hbaparams get` command shows port-level HBA parameters information for the Marvell Adapter port. This command is supported only on the physical port (not on the NPIV virtual port).

~# esxcli qlfc qcc port hbaparams get

Syntax

`esxcli qlfc qcc port hbaparams get [cmd options]`

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port hbaparams get` command on a local host:

~# esxcli qlfc qcc port hbaparams get -i FC_51:40:2e:c0:12:3c:f4:a0

The following example shows the `port hbaparams get` command on a remote host:

 # esxcli -s 172.27.9.144 -u root -p password qlfc qcc port hbaparams get -i FC_51:40:2e:c0:12:3c:f4:a0

The output for both commands is:

```
HBA Parameters Info:
   HBAParameters:
        Connection Options:Loop_Preferred_Otherwise_Point_To_Point
         Data Rate: Auto
         Enable Fabric Assigned WWN: false
         Enable Fc Tape Support: true
         Enable Hard Loop Id: true
         Enable LR: false
         Enable Lip Full Login: true
         Enable USCM: true
         Enable Target Reset: true
```

```
            Execution Throttle: 0
            Frame Size: 2048
            Hard Loop Id: 10
            Interrupt Delay Timer In100us: 2
            Link Down Timeout In Seconds: 30
            Login Retry Count: 8
            Loop Reset Delay In Seconds: 15
            Luns Per Target: 128
            Operation Mode: Interrupt_When_Interrupt_Delay_Timer_
             Expires_Or_No_Active_I/O
            Port Down Retry Count: 30
            Virtual Lane: false
      Message: Success
```

SetFile Port HBA Parameters

The `port hbaparams setFile` command sets port-level HBA parameters information for the specified Marvell Adapter port in a local cache file. This cache file is used by the `set` command to write HBA parameters in the flash. The `port hbaparams setFile` command is supported only on the physical port (not on the NPIV virtual port).

`~# esxcli qlfc qcc port hbaparams setFile`

Syntax

`esxcli qlfc qcc port hbaparams setFile [cmd options]`

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

`-c|--connection-options <str>` = Connection options

`-d|--data-rate <str>` = Data rate

`-b|--enable-bios <str>` = Enable BIOS (use for boot from SAN)

`-n|--enable-fabric-assigned-wwn <str>` = Enable fabric assigned WWN

`-f|--enable-fc-tape <str>` = Enable FC tape

`-p|--enable-hard-loop-id <str>` = Enable hard loop ID

`-g|--enable-lip-full-login <str>` = Enable LIP full login

`-k|--enable-lr <str>` = Enable LR

`-q|--enable-uscm <str>` = Enable Universal SAN Congestion Mitigation (USCM)

`-a|--enable-target-reset <str>` = Enable target reset

`-x|--execution-throttle <long>` = Execution throttle

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

`-z|--frame-size <long>` = Frame size

`-l|--hard-loop-id <long>` = Hard loop ID

`-e|--interrupt-delay-timer <long>` = Interrupt delay timer

`-o|--link-down-timeout <long>` = Link down time-out

`-u|--login-retry-count <long>` = Login retry count

`-r|--loop-reset-delay <long>` = Loop reset delay

`-y|--luns-per-target <long>` = Maximum LUNs per target

`-w|--port-down-retry-count <long>` = Port down retry count

`-m|--operation-mode <str>` = Operation mode

`-j|--virtual-lane <str>` = Enable virtual lane
For more information about virtual lanes, see the appropriate Marvell QLogic Fibre Channel Adapter user's guide.

Parameter	**Value (strings are case-sensitive)**
`identifier`	Port identifier from the port `list` command
`connection-options`	Loop_Only, Point_To_Point_Only, Loop_Preferred_Otherwise_Point_To_Point
`data-rate`	Auto, 1_Gbps, 2_Gbps, 4_Gbps, 8_Gbps, 16_Gbps, 32_Gbps, 64_Gbps [a]
`enable-bios`	true, false
`enable-fabric-assigned-wwn`	true, false
`enable-fc-tape`	true, false
`enable-hard-loop-id`	true, false
`enable-lip-full-login`	true, false
`enable-lr`	true, false
`enable-uscm`	true, false
`enable-target-reset`	true, false
`execution-throttle`	`0–65535`
`frame-size`	`512`, `1024`, `2048`, `2112` [b]
`hard-loop-id`	`0–125`
`interrupt-delay-timer`	`0–255`

Parameter	Value (strings are case-sensitive)
link-down-timeout	0–240
login-retry-count	0–255
loop-reset-delay	0–255
luns-per-target	0, 8, 16, 32, 64, 128, 256
port-down-retry-count	0–255
operation-mode	Interrupt_For_Every_I/O_Completion, Interrupt_When_Interrupt_Delay_Timer_Expires, Interrupt_When_Interrupt_Delay_Timer_ Expires_Or_No_Active_I/O
virtual-lane	true, false For more information about virtual lanes, see the appropriate Marvell QLogic Fibre Channel Adapter user's guide.

[a] 2800 Series Adapters only. In addition, the Loop_Only connection option is not supported for 64Gbps adapters.

[b] Frame size 2112 is supported only on 2880 Series Adapters.

Example

The following example shows the `port hbaparams setFile` command on a local host:

```
~# esxcli qlfc qcc port hbaparams setFile -i
FC_51:40:2e:c0:12:3c:f4:a0 -z 2048
```

The following example shows the `port hbaparams setFile` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
hbaparams setFile -i FC_51:40:2e:c0:12:3c:f4:a0 -z 2048
```

The output for both commands is:

```
HBA Parameters Info:
   HBAParameters:
         Connection Options:Loop_Preferred_Otherwise_Point_
                            To_Point
         Data Rate: Auto
         Enable Fabric Assigned WWN: false
         Enable Fc Tape Support: true
         Enable Hard Loop Id: true
         Enable Host Hba Bios: true
         Enable LR: false
         Enable Lip Full Login: true
```

```
            Enable USCM: true
            Enable Target Reset: true
            Execution Throttle: 0
            Frame Size: 2048
            Hard Loop Id: 10
            Interrupt Delay Timer In100us: 2
            Link Down Timeout In Seconds: 30
            Login Retry Count: 8
            Loop Reset Delay In Seconds: 15
            Luns Per Target: 128
            Operation Mode:
Interrupt_When_Interrupt_Delay_Timer_Expires_Or_No_Active_I/O
            Port Down Retry Count: 30
            Virtual Lane: false
   Message: Success
```

Set Port HBA Parameters

The `port hbaparams set` command sets port-level HBA parameters information for the Marvell Adapter port from the local cache file. If the local cache file is not present, this command fails. When this command completes successfully, the local cache file is deleted. This command is supported only on the physical port (not on the NPIV virtual port).

~# esxcli qlfc qcc port hbaparams set

Syntax

`esxcli qlfc qcc port hbaparams set [cmd options]`

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port hbaparams set` command on a local host:

~# esxcli qlfc qcc port hbaparams set -i FC_51:40:2e:c0:12:3c:f4:a0

The following example shows the `port hbaparams set` command on a remote host:

 # esxcli -s 172.27.9.144 -u root -p password qlfc qcc port hbaparams set -i FC_51:40:2e:c0:12:3c:f4:a0

The output for both commands is:

```
HBA Parameters Info:
   HBAParameters:
        Connection Options:Loop_Preferred_Otherwise_Point_To_Point
         Data Rate: Auto
         Enable Fabric Assigned WWN: false
         Enable Fc Tape Support: true
         Enable Hard Loop Id: true
         Enable LR: false
         Enable Lip Full Login: true
         Enable USCM: true
         Enable Target Reset: true
         Execution Throttle: 0
         Frame Size: 2048
         Hard Loop Id: 10
         Interrupt Delay Timer In100us: 2
         Link Down Timeout In Seconds: 30
         Login Retry Count: 8
         Loop Reset Delay In Seconds: 15
         Luns Per Target: 128
         Operation Mode:
     Interrupt_When_Interrupt_Delay_Timer_Expires_Or_No_Active_I/O
         Port Down Retry Count: 30
         Virtual Lane: false
Message: Success
   Reboot Required: true
```

Restore Default Port HBA Parameters

The `port hbaparams resetoreDefaults` command sets port-level HBA parameters information for the specified Marvell Adapter port to its default values. This command also deletes all the boot settings. This command is supported only on the physical port (not on the NPIV virtual port).

`~# esxcli qlfc qcc port hbaparams restoreDefaults`

Syntax

```
esxcli qlfc qcc port hbaparams restoreDefaults [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port hbaparams restoreDefaults` command on a local host:

```
~# esxcli qlfc qcc port hbaparams restoreDefaults -i
FC_51:40:2e:c0:12:3c:f4:a0
```

The following example shows the `port hbaparams restoreDefaults` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
hbaparams restoreDefaults -i FC_51:40:2e:c0:12:3c:f4:a0
```

The output for both commands is:

```
Restore Defaults Info:
   Message: Success
   Reboot Required: true
```

Save Port HBA Parameters

The `port hbaparams save` command saves port-level HBA parameters information for the specified port on the Marvell Adapter to a user-provided file. This command is supported on the physical port (not on an NPIV virtual port).

```
~# esxcli qlfc qcc port hbaparams save
```

Syntax

```
esxcli qlfc qcc port hbaparams save [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

`-f|--filename <str>` = File where Marvell Adapter port parameters will be saved. The file is in text format.

Example

The following example shows the `port hbaparams save` command on a local host:

```
~# esxcli qlfc qcc port hbaparams save -i
FC_51:40:2e:c0:12:3c:f4:a0 -f fc_params.dat
```

The following example shows the `port hbaparams save` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
hbaparams save -i FC_51:40:2e:c0:12:3c:f4:a0 -f fc_params.dat
```

The output for both commands is:

```
HBA Parameters Save:
   Message: Success
```

Adapter Port Buffer-To-Buffer Credit Recovery Configuration

The `port bbcr` command configures buffer-to-buffer credit (BBCR) values of the selected Marvell Adapter port:

```
~# esxcli qlfc qcc port bbcr
```

Syntax

```
esxcli qlfc qcc port bbcr {cmd} [cmd options]
```

Available Commands

`get`
Shows BBCR configuration of the specified Marvell Adapter port. See Get Port BBCR Configuration.

`setFile`
Sets BBCR configuration of the specified Marvell Adapter port in a local cache file. See SetFile Port BBCR Configuration.

`set`
Sets BBCR configuration of the specified Marvell Adapter port from a local cache file to flash on the adapter. See Set Port BBCR Configuration.

Get Port BBCR Configuration

The `port bbcr get` command shows port-level BBCR configuration for the specified Marvell Adapter port. This command is supported only on the physical port (not on the NPIV virtual port).

```
~# esxcli qlfc qcc port bbcr get
```

Syntax

```
esxcli qlfc qcc port bbcr get [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port bbcr get` command on a local host:

```
~# esxcli qlfc qcc port bbcr get -i FC_51:40:2x1e:c0:12:3c:f4:a0
```

The following example shows the `port bbcr get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port bbcr get
-i FC_51:40:2e:c0:12:3c:f4:a0
```

The output for both commands is:

```
BBCR Info:
   BBCR:
         Data:
      BBCR Status: true
      BBCR Configured: 4
      BBCR State: Online
      BBCR Negotiated: 4

   Message: Success
```

SetFile Port BBCR Configuration

The `port bbcr setFile` command sets port-level BBCR configuration for the specified Marvell Adapter port in a local cache file. This cache file is used by the `set` command to write BBCR configuration in the flash. This command is supported only on the physical port (not on the NPIV virtual port).

```
~# esxcli qlfc qcc port bbcr setFile
```

Syntax

```
esxcli qlfc qcc port bbcr setFile [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

`-c|--bbcr-configured <long>` = Buffer-to-buffer credit configured value

`-b|--bbcr-status <str>` = Buffer-to-buffer credit status

Parameter	Value (strings are case-sensitive)
Identifier	Port identifier from the port list command
bbcr-configured	0–15
bbcr-status	true, false

Example

The following example shows the port bbcr setFile command on a local host:

```
~# esxcli qlfc qcc port bbcr setFile -i FC_51:40:2e:c0:12:3c:f4:a0
-c 8
```

The following example shows the port hbaparams setFile command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port bbcr
setFile -i FC_51:40:2e:c0:12:3c:f4:a0 -c 8
```

The output for both commands is:

```
BBCR Info:
   BBCR:
         Data:
      BBCR Status: true
      BBCR Configured: 8
      BBCR State: Online
      BBCR Negotiated: 4


   Message: Success
```

Set Port BBCR Configuration

The port bbcr set command sets port-level BBCR configuration for the specified Marvell Adapter port from the local cache file. If the local cache file is not present, this command fails. When this command completes successfully, the local cache file is deleted. This command is supported only on the physical port (not on the NPIV virtual port).

```
~# esxcli qlfc qcc port bbcr set
```

Syntax

```
esxcli qlfc qcc port bbcr set [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port bbcr set` command on a local host:

```
~# esxcli qlfc qcc port bbcr set -i FC_51:40:2e:c0:12:3c:f4:a0
```

The following example shows the `port bbcr set` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port bbcr set
-i FC_51:40:2e:c0:12:3c:f4:a0
```

The output for both commands is:

```
BBCR Info:
   BBCR:
         Data:
      BBCR Status: true
      BBCR Configured: 8
      BBCR State: Online
      BBCR Negotiated: 4

   Message: Success
   Reboot Required: true
```

Adapter Port Forward-Error-Correction Configuration

The `port fec` command configures forward-error-correction (FEC) values of the selected Marvell Adapter port:

```
~# esxcli qlfc qcc port fec
```

Syntax

```
esxcli qlfc qcc port fec {cmd} [cmd options]
```

Available Commands

`get`
Shows FEC configuration of the specified Marvell Adapter port. See Get Port FEC Configuration.

`set`
Sets FEC configuration of the specified Marvell Adapter port to flash on the adapter. See Set Port FEC Configuration.

Get Port FEC Configuration

The `port fec get` command shows port-level FEC configuration for the specified Marvell Adapter port. This command is supported only on the physical port (not on the NPIV virtual port).

~# esxcli qlfc qcc port fec get

Syntax

```
esxcli qlfc qcc port fec get [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port fec get` command on a local host:

~# esxcli qlfc qcc port fec get -i FC_51:40:2e:c0:12:3c:f4:a0

The following example shows the `port fec get` command on a remote host:

 # esxcli -s 172.27.9.144 -u root -p password qlfc qcc port fec get -i FC_51:40:2e:c0:12:3c:f4:a0

The output for both commands is:

```
FEC Info:
   FEC:
         Correctable FECerrors: 0
         FEC_Status: true
         Uncorrectable FECerrors: 0
   Message: Success
```

Set Port FEC Configuration

The `port fec set` command sets port-level FEC configuration for the specified Marvell Adapter port. This command is supported only on the physical port (not on the NPIV virtual port).

~# esxcli qlfc qcc port fec set

Syntax

```
esxcli qlfc qcc port fec set [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

`-f|--fec-status <str>` = FEC status

`-r|--reset <str>` = FEC counter reset

Parameter	Value (strings are case-sensitive)
`Identifier`	Port identifier from the `port list` command
`fec-status`	true, false
`reset`	true, false

Example

The following example shows the `port fec set` command on a local host:

```
~# esxcli qlfc qcc port fec set -i FC_51:40:2e:c0:12:3c:f4:a0 -f true
```

The following example shows the `port fec set` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port fec set
-i FC_51:40:2e:c0:12:3c:f4:a0 -f true
```

The output for both commands is:

```
FEC Info:
   FEC:
         Correctable FECerrors: 0
         FEC_Status: true
         Uncorrectable FECerrors: 0
   Message: Success
   Reboot Required: true
```

Adapter Port Boot Configuration

The `port bootconfig` command configures boot target and LUN settings for boot from SAN of the selected Marvell Adapter port:

```
~# esxcli qlfc qcc port bootconfig
```

Syntax

```
esxcli qlfc qcc port bootconfig {cmd} [cmd options]
```

Available Commands

`get`
Shows the boot configuration of the specified Marvell Adapter port. See Get Port Boot Configuration.

`setFile`
Sets the boot configuration of the specified Marvell Adapter port in a local cache file. See SetFile Port Boot Configuration.

`set`
Sets the boot configuration of the specified Marvell Adapter port from a local cache file to flash on the adapter. See Set Port Boot Configuration.

Get Port Boot Configuration

The `port bootconfig get` command shows port-level boot configuration for the specified Marvell Adapter port. This command is supported only on the physical port (not on the NPIV virtual port).

```
~# esxcli qlfc qcc port bootconfig get
```

Syntax

```
esxcli qlfc qcc port bootconfig get [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

`-b|--boot-mode <str>` = Boot mode (BIOS, UEFI). UEFI mode is supported only on adapters with ISP2700/2800 Series Controllers.

Example

The following example shows the `port bootconfig get` command on a local host in BIOS mode:

```
~# esxcli qlfc qcc port bootconfig get -i FC_51:40:2e:c0:12:3c:f4:a0
-b BIOS
```

The following example shows the `port bootconfig get` command on a remote host in BIOS mode:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
bootconfig get -i FC_51:40:2e:c0:12:3c:f4:a0 -b BIOS
```

The output for both commands is:

```
Boot Config Info:
   Boot Config:
         Data:
      Adapter BIOS: false
```

```
        Selectable Boot: false
        Fabric Assigned Boot LUN: true
        Drive 0 WWPN: 00:00:00:00:00:00:00:00
        Drive 0 LUN: 0
        Drive 1 WWPN: 00:00:00:00:00:00:00:00
        Drive 1 LUN: 0
        Drive 2 WWPN: 00:00:00:00:00:00:00:00
        Drive 2 LUN: 0
        Drive 3 WWPN: 00:00:00:00:00:00:00:00
        Drive 3 LUN: 0
Message: Success
```

NOTE

UEFI mode is supported only for ISP2700 and ISP2800 based adapters

The following example shows the `port bootconfig get` command on a local host in UEFI mode:

```
~# esxcli qlfc qcc port bootconfig get -i FC_51:40:2e:c0:12:3c:f4:a0
-b UEFI
```

The following example shows the `port bootconfig get` command on a remote host in UEFI mode:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
bootconfig get -i FC_51:40:2e:c0:12:3c:f4:a0 -b UEFI
```

The output for both commands is:

```
     Data:
Adapter Driver: false
Selective Login: false
Selective LUN Login: false
World Login: true
Fabric Assigned Boot LUN: false
Fabric Assigned Boot LUN: false
Drive 0 WWPN: 00:00:00:00:00:00:00:00
Drive 0 LUN: 0
Drive 1 WWPN: 00:00:00:00:00:00:00:00
Drive 1 LUN: 0
Drive 2 WWPN: 00:00:00:00:00:00:00:00
Drive 2 LUN: 0
Drive 3 WWPN: 00:00:00:00:00:00:00:00
```

```
Drive 3 LUN: 0
Drive 4 WWPN: 00:00:00:00:00:00:00:00
Drive 4 LUN: 0
Drive 5 WWPN: 00:00:00:00:00:00:00:00
Drive 5 LUN: 0
Drive 6 WWPN: 00:00:00:00:00:00:00:00
Drive 6 LUN: 0
Drive 7 WWPN: 00:00:00:00:00:00:00:00
Drive 7 LUN: 0
Message: Success
```

SetFile Port Boot Configuration

The `port bootconfig setFile` command sets port-level boot configuration for the specified Marvell Adapter port in a local cache file. This cache file is used by the `set` command to write the boot configuration in the flash. This command is supported only on the physical port (not on the NPIV virtual port).

`~# esxcli qlfc qcc port bootconfig setFile`

Syntax

`esxcli qlfc qcc port bootconfig setFile [cmd options]`

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

`-a|--enable-bfs<str>` = Enable boot-from-SAN

`-b|--boot-mode <str>` = Boot mode

`-c|--selective-lun-login <str>` = Selective LUN login (UEFI mode only)

`-d|--selectable-boot <str>` = Selectable boot (BIOS mode only)

`-e|--selective-login <str>` = Selective login (UEFI mode only)

`-f|--fabric-assigned-boot-lun <str>` = Fabric assigned boot LUN

`g|--world-login <str>` = World login (UEFI mode only)

`j|--drive-0 <str>` = Boot target 0 (“**:**” separated) and LUN number (“**-**” separated)

`k|--drive-1 <str>` = Boot target 1 (“**:**” separated) and LUN number (“**-**” separated)

`-l|--drive-2 <str>` = Boot target 2 (“**:**” separated) and LUN number (“**-**” separated)

`-m|--drive-3 <str>` = Boot target 3 (“**:**” separated) and LUN number (“**-**” separated)

`-n|--drive-4 <str>` = Boot target 4 (“**:**” separated) and LUN number (“**-**” separated) (UEFI mode only)

`-o|--drive-5 <str>` = Boot target 5 (“**:**” separated) and LUN number (“**-**” separated) (UEFI mode only)

`-p|--drive-6 <str>` = Boot target 6 (“**:**” separated) and LUN number (“**-**” separated) (UEFI mode only)

`-q|--drive-7 <str>` = Boot target 7 (“**:**” separated) and LUN number (“**-**” separated) (UEFI mode only)

Parameter	**Value (strings are case-sensitive)**
`Identifier`	Port identifier from the `port list` command
`boot-mode`	BIOS, UEFI
`enable-bfs`	true, false
`selectable-boot`	true, false
`fabric-assigned-boot-lun`	true, false
`selective-login`	true, false
`selective-lun-login`	true, false
`world-login`	true, false
`drive-0`	“**:**” separated WWPN and “**-**” separated LUN
`drive-1`	“**:**” separated WWPN and “**-**” separated LUN
`drive-2`	“**:**” separated WWPN and “**-**” separated LUN
`drive-3`	“**:**” separated WWPN and “**-**” separated LUN
`drive-4`	“**:**” separated WWPN and “**-**” separated LUN
`drive-5`	“**:**” separated WWPN and “**-**” separated LUN
`drive-6`	“**:**” separated WWPN and “**-**” separated LUN
`drive-7`	“**:**” separated WWPN and “**-**” separated LUN

Example

The following example shows the `port bootconfig setFile` command on a local host in BIOS mode:

```
~# esxcli qlfc qcc port bootconfig setFile -i
FC_51:40:2e:c0:12:3c:f4:a0 -b BIOS -d true
```

The following example shows the `port bootconfig setFile` command on a remote host in BIOS mode:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
bootconfig setFile -i FC_51:40:2e:c0:12:3c:f4:a0 -b BIOS -d true
```

The output for both commands is:

```
Boot Config Info:
   Boot Config:
         Data:
      Adapter BIOS: false
      Selectable Boot: true
      Fabric Assigned Boot LUN: true
      Drive 0 WWPN: 00:00:00:00:00:00:00:00
      Drive 0 LUN: 0
      Drive 1 WWPN: 00:00:00:00:00:00:00:00
      Drive 1 LUN: 0
      Drive 2 WWPN: 00:00:00:00:00:00:00:00
      Drive 2 LUN: 0
      Drive 3 WWPN: 00:00:00:00:00:00:00:00
      Drive 3 LUN: 0
Message: Success
```

The following example shows the `port bootconfig setFile` command on a local host in UEFI mode:

```
~# esxcli qlfc qcc port bootconfig setFile -i
FC_51:40:2e:c0:12:3c:f4:a0 -b UEFI -e true
```

The following example shows the `port bootconfig setFile` command on a remote host in UEFI mode:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
bootconfig setFile -i FC_51:40:2e:c0:12:3c:f4:a0 -b UEFI -e true
```

The output for both commands is:

```
Boot Config Info:
   Boot Config:
         Data:
      Adapter Driver: false
      Selective Login: true
      Selective LUN Login: false
      World Login: true
      Fabric Assigned Boot LUN: false
      Drive 0 WWPN: 00:00:00:00:00:00:00:00
```

```
        Drive 0 LUN: 0
        Drive 1 WWPN: 00:00:00:00:00:00:00:00
        Drive 1 LUN: 0
        Drive 2 WWPN: 00:00:00:00:00:00:00:00
        Drive 2 LUN: 0
        Drive 3 WWPN: 00:00:00:00:00:00:00:00
        Drive 3 LUN: 0
        Drive 4 WWPN: 00:00:00:00:00:00:00:00
        Drive 4 LUN: 0
        Drive 5 WWPN: 00:00:00:00:00:00:00:00
        Drive 5 LUN: 0
        Drive 6 WWPN: 00:00:00:00:00:00:00:00
        Drive 6 LUN: 0
        Drive 7 WWPN: 00:00:00:00:00:00:00:00
        Drive 7 LUN: 0

    Message: Success
```

Set Port Boot Configuration

The `port bootconfig set` command sets port-level boot configuration for the specified Marvell Adapter port from the local cache file. If the local cache file is not present, this command fails. When this command completes successfully, the local cache file is deleted.

```
~# esxcli qlfc qcc port bootconfig set
```

NOTE

This command is supported only on the physical port (not on the NPIV virtual port).

This command is supported only for Fibre Channel Protocol (FCP) targets. Non-Volatile Memory Express (NVMe) targets are not supported.

Syntax

```
esxcli qlfc qcc port bootconfig set [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port bootconfig set` command on a local host:

```
~# esxcli qlfc qcc port bootconfig set -i FC_51:40:2e:c0:12:3c:f4:a0
```

The following example shows the `port bootconfig set` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
bootconfig set -i FC_51:40:2e:c0:12:3c:f4:a0
```

The output for both commands depends on the last boot mode value used in the `setFile` command.

When the boot mode is BIOS, the command output is:

```
Boot Config Info:
   Boot Config:
      Data:
      Adapter BIOS: false
      Selectable Boot: true
      Fabric Assigned Boot LUN: true
      Drive 0 WWPN: 00:00:00:00:00:00:00:00
      Drive 0 LUN: 0
      Drive 1 WWPN: 00:00:00:00:00:00:00:00
      Drive 1 LUN: 0
      Drive 2 WWPN: 00:00:00:00:00:00:00:00
      Drive 2 LUN: 0
      Drive 3 WWPN: 00:00:00:00:00:00:00:00
      Drive 3 LUN: 0
Message: Success
Reboot Required: false
```

When the boot mode is UEFI, the command output is:

```
Boot Config Info:
   Boot Config:
      Data:
      Adapter Driver: false
      Selective Login: true
```

```
        Selective LUN Login: false
        World Login: true
        Fabric Assigned Boot LUN: false
        Drive 0 WWPN: 00:00:00:00:00:00:00:00
        Drive 0 LUN: 0
        Drive 1 WWPN: 00:00:00:00:00:00:00:00
        Drive 1 LUN: 0
        Drive 2 WWPN: 00:00:00:00:00:00:00:00
        Drive 2 LUN: 0
        Drive 3 WWPN: 00:00:00:00:00:00:00:00
        Drive 3 LUN: 0
        Drive 4 WWPN: 00:00:00:00:00:00:00:00
        Drive 4 LUN: 0
        Drive 5 WWPN: 00:00:00:00:00:00:00:00
        Drive 5 LUN: 0
        Drive 6 WWPN: 00:00:00:00:00:00:00:00
        Drive 6 LUN: 0
        Drive 7 WWPN: 00:00:00:00:00:00:00:00
        Drive 7 LUN: 0

    Message: Success
    Reboot Required: false
9
```

Adapter Port Boot Configuration (Extended)

The `port bootconfigext` command configures boot target and LUN settings to boot from SAN from the selected Marvell Adapter port:

```
~# esxcli qlfc qcc port bootconfigext
```

Syntax

```
esxcli qlfc qcc port bootconfigext {cmd} [cmd options]
```

Available Commands

`get`
Shows the boot configuration of the specified Marvell Adapter port. See Get Port Boot Configuration (Extended).

`setFile`
Sets the boot configuration of the specified Marvell Adapter port from a local cache file. See SetFile Port Boot Configuration (Extended).

`set`
Sets the boot configuration of the specified Marvell Adapter port from a local cache file to flash on the adapter. See Set Port Boot Configuration (Extended).

Get Port Boot Configuration (Extended)

The `port bootconfigext get` command shows port-level boot configuration for the specified Marvell Adapter port. This command is supported only on the physical port (not on the NPIV virtual port).

~# esxcli qlfc qcc port bootconfigext get

Syntax

`esxcli qlfc qcc port bootconfigext get [cmd options]`

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

`-b|--boot-mode <str>` = Boot mode (BIOS, UEFI, NVME). UEFI and NVME modes are supported only on adapters with ISP2700/2800 Series Controllers. This keyword is required.

Example

The following example shows the `port bootconfigext get` command on a local host in BIOS mode:

~# esxcli qlfc qcc port bootconfigext get -i FC_51:40:2e:c0:12:3c:f4:a0 -b BIOS

The following example shows the `port bootconfigext get` command on a remote host in BIOS mode:

 # esxcli -s 172.27.9.144 -u root -p password qlfc qcc port bootconfigext get -i FC_51:40:2e:c0:12:3c:f4:a0 -b BIOS

The output for both commands is:

```
Boot Config Info:
   Boot Config:
         Data:
      Adapter BIOS: false
      Selectable Boot: false
      Fabric Assigned Boot LUN: true
      Drive 0 WWPN: 00:00:00:00:00:00:00:00
      Drive 0 LUN: 0
      Drive 1 WWPN: 00:00:00:00:00:00:00:00
      Drive 1 LUN: 0
```

```
      Drive 2 WWPN: 00:00:00:00:00:00:00:00
      Drive 2 LUN: 0
      Drive 3 WWPN: 00:00:00:00:00:00:00:00
      Drive 3 LUN: 0
   Message: Success
```

The following example shows the `port bootconfigext get` command on a local host in UEFI mode:

```
~# esxcli qlfc qcc port bootconfigext get -i
FC_51:40:2e:c0:12:3c:f4:a0 -b UEFI
```

The following example shows the `port bootconfigext get` command on a remote host in UEFI mode:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
bootconfigext get -i FC_51:40:2e:c0:12:3c:f4:a0 -b UEFI
```

The output for both commands is:

```
Boot Config Info:
   Boot Config:
         Data:
      Adapter Driver: false
      Selective Login: false
      Selective LUN Login: false
      World Login: true
      Fabric Assigned Boot LUN: false
      Drive 0 WWPN: 00:00:00:00:00:00:00:00
      Drive 0 LUN: 0
      Drive 1 WWPN: 00:00:00:00:00:00:00:00
      Drive 1 LUN: 0
      Drive 2 WWPN: 00:00:00:00:00:00:00:00
      Drive 2 LUN: 0
      Drive 3 WWPN: 00:00:00:00:00:00:00:00
      Drive 3 LUN: 0
      Drive 4 WWPN: 00:00:00:00:00:00:00:00
      Drive 4 LUN: 0
      Drive 5 WWPN: 00:00:00:00:00:00:00:00
      Drive 5 LUN: 0
      Drive 6 WWPN: 00:00:00:00:00:00:00:00
      Drive 6 LUN: 0
```

```
      Drive 7 WWPN: 00:00:00:00:00:00:00:00
      Drive 7 LUN: 0

   Message: Success
```

The following example shows the `port bootconfigext get` command on a local host operating in NVME mode:

```
~# esxcli qlfc qcc port bootconfigext get -i
FC_51:40:2e:c0:12:3c:f4:a0 -b NVME
```

The following example shows the `port bootconfigext get` command on a remote host in UEFI mode:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
bootconfigext get -i FC_51:40:2e:c0:12:3c:f4:a0 -b NVME
```

```
The output for both commands is:
Boot Config Info:
   Boot Config:
         Data:
      Adapter Driver: true
      Selective Login: false
      Selective LUN Login: false
      World Login: true
      Fabric Assigned Boot LUN: false
      FC NVMe: true
      Host NQN: nqn.2014-08.com.marvell:nvme:FCBFSAutomationLun0
      Host ID: 62978f65f57dabb29525f40270d6f210
      Storage 0: true
      Storage 0 WWNN: 23:98:00:a0:98:f4:68:4a
      Storage 0 WWPN: 23:9a:00:a0:98:f4:68:4a
      Storage 0 NQN: nqn.1992-08.com.netapp:sn.
                     4d9248d2e18011ecad6d00a098d6abfa:
                     subsystem.AM-Automation-SUB
      Storage 0 ControllerID: 0x101
      Storage 0 NamespaceID: 7
      Storage 1: false
      Storage 1 WWNN: 00:00:00:00:00:00:00:00
      Storage 1 WWPN: 00:00:00:00:00:00:00:00
      Storage 1 NQN:
      Storage 1 ControllerID: 0x0
      Storage 1 NamespaceID: 0
      Storage 2: false
```

```
    Storage 2 WWNN: 00:00:00:00:00:00:00:00
    Storage 2 WWPN: 00:00:00:00:00:00:00:00
    Storage 2 NQN:
    Storage 2 ControllerID: 0x0
    Storage 2 NamespaceID: 0
    Storage 3: false
    Storage 3 WWNN: 00:00:00:00:00:00:00:00
    Storage 3 WWPN: 00:00:00:00:00:00:00:00
    Storage 3 NQN:
    Storage 3 ControllerID: 0x0
    Storage 3 NamespaceID: 0
    Storage 4: false
    Storage 4 WWNN: 00:00:00:00:00:00:00:00
    Storage 4 WWPN: 00:00:00:00:00:00:00:00
    Storage 4 NQN:
    Storage 4 ControllerID: 0x0
    Storage 4 NamespaceID: 0
    Storage 5: false
    Storage 5 WWNN: 00:00:00:00:00:00:00:00
    Storage 5 WWPN: 00:00:00:00:00:00:00:00
    Storage 5 NQN:
    Storage 5 ControllerID: 0x0
    Storage 5 NamespaceID: 0
    Storage 6: false
    Storage 6 WWNN: 00:00:00:00:00:00:00:00
    Storage 6 WWPN: 00:00:00:00:00:00:00:00
    Storage 6 NQN:
    Storage 6 ControllerID: 0x0
    Storage 6 NamespaceID: 0
    Storage 7: false
    Storage 7 WWNN: 00:00:00:00:00:00:00:00
    Storage 7 WWPN: 00:00:00:00:00:00:00:00
    Storage 7 NQN:
    Storage 7 ControllerID: 0x0
    Storage 7 NamespaceID: 0

Message: Success
```

SetFile Port Boot Configuration (Extended)

The `port bootconfigext setFile` command sets port-level boot configuration for the specified Marvell Adapter port in a local cache file. This cache file is used by the `set` command to write the boot configuration in the flash. This command is supported only on the physical port (not on the NPIV virtual port).

`~# esxcli qlfc qcc port bootconfigext setFile`

Syntax

```
esxcli qlfc qcc port bootconfigext setFile [cmd options]
```

Keywords

```
                -i|--identifier <str> = Port identifier (required). Obtain the
                                        identifier from the port list output.
                 -a|--enable-bfs<str> = Enable boot-from-SAN
                 -b|--boot-mode <str> = Boot mode
    -c|--selective-lun-login <str> = Selective LUN login (UEFI and NVME
                                        modes only)
        -d|--selectable-boot <str> = Selectable boot (BIOS mode only)
        -e|--selective-login <str> = Selective login (UEFI and NVMe modes
                                        only)
-f|--fabric-assigned-boot-lun <str> = Fabric assigned boot LUN
              g|--world-login <str> = World login (UEFI and NVMe modes only)
     -l|--boot-target-enable <str> = Boot target enable (NVMe mode only)
                 -m|--fc-nvme <str> = Fibre Channel NVMe (NVMe mode only)
       -n|--boot-target-num <long> = Boot target number
         -o|--boot-disk-num <long> = Boot disk (LUN or namespace ID) number
       -p|--boot-target-wwpn <str> = Boot target WWPN (":" separated)
```

Parameter	Value (strings are case-sensitive)
`Identifier`	Port identifier from the `port list` command
`boot-mode`	BIOS, UEFI, NVME
`enable-bfs`	true, false
`selectable-boot`	true, false
`fabric-assigned-boot-lun`	true, false
`selective-login`	true, false
`selective-lun-login`	true, false

Parameter	Value (strings are case-sensitive)
world-login	true, false
fc-nvme	true, false
boot-target-num	0–3 (BIOS) or 0–7 (UEFI mode and NVMe mode)
boot-target-wwpn	"**:**" separated WWPN
boot-disk-num	0–65,535
boot-target-enable	true, false

Example

The following example shows the `port bootconfigext setFile` command on a local host in BIOS mode:

```
~# esxcli qlfc qcc port bootconfigext setFile -i
FC_51:40:2e:c0:12:3c:f4:a0 -b BIOS -d true
```

The following example shows the `port bootconfigext setFile` command on a remote host in BIOS mode:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
bootconfigext setFile -i FC_51:40:2e:c0:12:3c:f4:a0 -b BIOS -d
true
```

The output for both commands is:

```
Boot Config Info:
   Boot Config:
         Data:
      Adapter BIOS: false
      Selectable Boot: true
      Fabric Assigned Boot LUN: true
      Drive 0 WWPN: 00:00:00:00:00:00:00:00
      Drive 0 LUN: 0
      Drive 1 WWPN: 00:00:00:00:00:00:00:00
      Drive 1 LUN: 0
      Drive 2 WWPN: 00:00:00:00:00:00:00:00
      Drive 2 LUN: 0
      Drive 3 WWPN: 00:00:00:00:00:00:00:00
      Drive 3 LUN: 0
   Message: Success
```

The following example shows the `port bootconfigext setFile` command on a local host in UEFI mode:

```
~# esxcli qlfc qcc port bootconfigext setFile -i
FC_51:40:2e:c0:12:3c:f4:a0 -b UEFI -e true
```

The following example shows the `port bootconfigext setFile` command on a remote host in UEFI mode:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
bootconfigext setFile -i FC_51:40:2e:c0:12:3c:f4:a0 -b UEFI -e
true
```

```
The output for both commands is:
Boot Config Info:
   Boot Config:
         Data:
      Adapter Driver: false
      Selective Login: true
      Selective LUN Login: false
      World Login: true
      Fabric Assigned Boot LUN: false
      Drive 0 WWPN: 00:00:00:00:00:00:00:00
      Drive 0 LUN: 0
      Drive 1 WWPN: 00:00:00:00:00:00:00:00
      Drive 1 LUN: 0
      Drive 2 WWPN: 00:00:00:00:00:00:00:00
      Drive 2 LUN: 0
      Drive 3 WWPN: 00:00:00:00:00:00:00:00
      Drive 3 LUN: 0
      Drive 4 WWPN: 00:00:00:00:00:00:00:00
      Drive 4 LUN: 0
      Drive 5 WWPN: 00:00:00:00:00:00:00:00
      Drive 5 LUN: 0
      Drive 6 WWPN: 00:00:00:00:00:00:00:00
      Drive 6 LUN: 0
      Drive 7 WWPN: 00:00:00:00:00:00:00:00
      Drive 7 LUN: 0

   Message: Success
```

The following example shows the `port bootconfigext setFile` command on a local host in NVME mode:

```
~# esxcli qlfc qcc port bootconfigext setFile -i
FC_51:40:2e:c0:12:3c:f4:a0 -b NVME -e true
```

The following example shows the `port bootconfigext setFile` command on a remote host in NVME mode:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
bootconfigext setFile -i FC_51:40:2e:c0:12:3c:f4:a0 -b NVME -e
true
```

The output for both commands is:

```
Boot Config Info:
   Boot Config:
         Data:
      Adapter Driver: true
      Selective Login: true
      Selective LUN Login: false
      World Login: true
      Fabric Assigned Boot LUN: false
      FC NVMe: true
      Host NQN: nqn.2014-08.com.marvell:nvme:FCBFSAutomationLun0
      Host ID: 62978f65f57dabb29525f40270d6f210
      Storage 0: true
      Storage 0 WWNN: 23:98:00:a0:98:f4:68:4a
      Storage 0 WWPN: 23:9a:00:a0:98:f4:68:4a
      Storage 0 NQN: nqn.1992-08.com.netapp:sn.
                     4d9248d2e18011ecad6d00a098d6abfa:
                     subsystem.AM-Automation-SUB
      Storage 0 ControllerID: 0x101
      Storage 0 NamespaceID: 7
      Storage 1: false
      Storage 1 WWNN: 00:00:00:00:00:00:00:00
      Storage 1 WWPN: 00:00:00:00:00:00:00:00
      Storage 1 NQN:
      Storage 1 ControllerID: 0x0
      Storage 1 NamespaceID: 0
      Storage 2: false
      Storage 2 WWNN: 00:00:00:00:00:00:00:00
      Storage 2 WWPN: 00:00:00:00:00:00:00:00
      Storage 2 NQN:
```

```
    Storage 2 ControllerID: 0x0
    Storage 2 NamespaceID: 0
    Storage 3: false
    Storage 3 WWNN: 00:00:00:00:00:00:00:00
    Storage 3 WWPN: 00:00:00:00:00:00:00:00
    Storage 3 NQN:
    Storage 3 ControllerID: 0x0
    Storage 3 NamespaceID: 0
    Storage 4: false
    Storage 4 WWNN: 00:00:00:00:00:00:00:00
    Storage 4 WWPN: 00:00:00:00:00:00:00:00
    Storage 4 NQN:
    Storage 4 ControllerID: 0x0
    Storage 4 NamespaceID: 0
    Storage 5: false
    Storage 5 WWNN: 00:00:00:00:00:00:00:00
    Storage 5 WWPN: 00:00:00:00:00:00:00:00
    Storage 5 NQN:
    Storage 5 ControllerID: 0x0
    Storage 5 NamespaceID: 0
    Storage 6: false
    Storage 6 WWNN: 00:00:00:00:00:00:00:00
    Storage 6 WWPN: 00:00:00:00:00:00:00:00
    Storage 6 NQN:
    Storage 6 ControllerID: 0x0
    Storage 6 NamespaceID: 0
    Storage 7: false
    Storage 7 WWNN: 00:00:00:00:00:00:00:00
    Storage 7 WWPN: 00:00:00:00:00:00:00:00
    Storage 7 NQN:
    Storage 7 ControllerID: 0x0
    Storage 7 NamespaceID: 0

Message: Success
```

Set Port Boot Configuration (Extended)

The `port bootconfigext set` command sets port-level boot configuration for the specified Marvell Adapter port from the local cache file. If the local cache file is not present, this command fails. When this command completes successfully, the local cache file is deleted.

```
~# esxcli qlfc qcc port bootconfigext set
```

NOTE

This command is supported only on the physical port (not on the NPIV virtual port).

This command is supported for both Fibre Channel Protocol (FCP) targets and non-Volatile Memory Express (NVMe) targets.

Syntax

```
esxcli qlfc qcc port bootconfigext set [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port bootconfigext set` command on a local host:

```
~# esxcli qlfc qcc port bootconfigext set -i
FC_51:40:2e:c0:12:3c:f4:a0
```

The following example shows the `port bootconfigext set` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
bootconfigext set -i FC_51:40:2e:c0:12:3c:f4:a0
```

The output for both commands is dependent on the last boot mode value used in the `setFile` command.

If BIOS boot mode is used, the output is:

```
Boot Config Info:
   Boot Config:
         Data:
      Adapter BIOS: false
      Selectable Boot: true
```

```
      Fabric Assigned Boot LUN: true
      Drive 0 WWPN: 00:00:00:00:00:00:00:00
      Drive 0 LUN: 0
      Drive 1 WWPN: 00:00:00:00:00:00:00:00
      Drive 1 LUN: 0
      Drive 2 WWPN: 00:00:00:00:00:00:00:00
      Drive 2 LUN: 0
      Drive 3 WWPN: 00:00:00:00:00:00:00:00
      Drive 3 LUN: 0
   Message: Success
   Reboot Required: false
```

If UEFI boot mode used, the output is:

```
Boot Config Info:
   Boot Config:
         Data:
      Adapter Driver: false
      Selective Login: true
      Selective LUN Login: false
      World Login: true
      Fabric Assigned Boot LUN: false
      Fabric Assigned Boot LUN: false
      Drive 0 WWPN: 00:00:00:00:00:00:00:00
      Drive 0 LUN: 0
      Drive 1 WWPN: 00:00:00:00:00:00:00:00
      Drive 1 LUN: 0
      Drive 2 WWPN: 00:00:00:00:00:00:00:00
      Drive 2 LUN: 0
      Drive 3 WWPN: 00:00:00:00:00:00:00:00
      Drive 3 LUN: 0
      Drive 4 WWPN: 00:00:00:00:00:00:00:00
      Drive 4 LUN: 0
      Drive 5 WWPN: 00:00:00:00:00:00:00:00
      Drive 5 LUN: 0
      Drive 6 WWPN: 00:00:00:00:00:00:00:00
      Drive 6 LUN: 0
      Drive 7 WWPN: 00:00:00:00:00:00:00:00
      Drive 7 LUN: 0
```

```
   Message: Success
   Reboot Required: false
```

If NVMe boot mode is used, the output is:

```
Boot Config Info:
   Boot Config:
         Data:
      Adapter Driver: true
      Selective Login: true
      Selective LUN Login: false
      World Login: true
      Fabric Assigned Boot LUN: false
      FC NVMe: true
      Host NQN: nqn.2014-08.com.marvell:nvme:FCBFSAutomationLun0
      Host ID: 62978f65f57dabb29525f40270d6f210
      Storage 0: true
      Storage 0 WWNN: 23:98:00:a0:98:f4:68:4a
      Storage 0 WWPN: 23:9a:00:a0:98:f4:68:4a
      Storage 0 NQN: nqn.1992-08.com.netapp:
                     sn.4d9248d2e18011ecad6d00a098d6abfa:
                     subsystem.AM-Automation-SUB
      Storage 0 ControllerID: 0x101
      Storage 0 NamespaceID: 7
      Storage 1: false
      Storage 1 WWNN: 00:00:00:00:00:00:00:00
      Storage 1 WWPN: 00:00:00:00:00:00:00:00
      Storage 1 NQN:
      Storage 1 ControllerID: 0x0
      Storage 1 NamespaceID: 0
      Storage 2: false
      Storage 2 WWNN: 00:00:00:00:00:00:00:00
      Storage 2 WWPN: 00:00:00:00:00:00:00:00
      Storage 2 NQN:
      Storage 2 ControllerID: 0x0
      Storage 2 NamespaceID: 0
      Storage 3: false
      Storage 3 WWNN: 00:00:00:00:00:00:00:00
      Storage 3 WWPN: 00:00:00:00:00:00:00:00
      Storage 3 NQN:
      Storage 3 ControllerID: 0x0
```

```
    Storage 3 NamespaceID: 0
    Storage 4: false
    Storage 4 WWNN: 00:00:00:00:00:00:00:00
    Storage 4 WWPN: 00:00:00:00:00:00:00:00
    Storage 4 NQN:
    Storage 4 ControllerID: 0x0
    Storage 4 NamespaceID: 0
    Storage 5: false
    Storage 5 WWNN: 00:00:00:00:00:00:00:00
    Storage 5 WWPN: 00:00:00:00:00:00:00:00
    Storage 5 NQN:
    Storage 5 ControllerID: 0x0
    Storage 5 NamespaceID: 0
    Storage 6: false
    Storage 6 WWNN: 00:00:00:00:00:00:00:00
    Storage 6 WWPN: 00:00:00:00:00:00:00:00
    Storage 6 NQN:
    Storage 6 ControllerID: 0x0
    Storage 6 NamespaceID: 0
    Storage 7: false
    Storage 7 WWNN: 00:00:00:00:00:00:00:00
    Storage 7 WWPN: 00:00:00:00:00:00:00:00
    Storage 7 NQN:
    Storage 7 ControllerID: 0x0
    Storage 7 NamespaceID: 0

Message: Success
Reboot Required: false
```

Adapter Port Firmware Dump Information

The `port fwdump` command provides firmware dump information from the selected Marvell Adapter port:

`~# esxcli qlfc qcc port fwdump`

Syntax

`esxcli qlfc qcc port fwdump {cmd} [cmd options]`

Available Commands

`get`
Saves a firmware dump of the specified Marvell Adapter port. See Get Port Firmware Dump Information.

Get Port Firmware Dump Information

The `port fwdump get` command saves port-level firmware dump information for the specified Marvell Adapter port. This command works only when the firmware dump is generated and saved in the driver buffer. This command is supported only on the physical port (not on the NPIV virtual port).

~# esxcli qlfc qcc port fwdump get

Syntax

```
esxcli qlfc qcc port fwdump get [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

`-f|--filename <str>` = Filename where the firmware dump will be saved (required) with the absolute path

Example

The following example shows the `port fwdump get` command on a local host:

```
~# esxcli qlfc qcc port fwdump get -i FC_51:40:2e:c0:12:3c:f4:a0
-f /scratch/fwdump.bin
```

The following example shows the `port fwdump get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port fwdump
get -i FC_51:40:2e:c0:12:3c:f4:a0 -f /scratch/fwdump.bin
```

The output for both commands is:

```
FwDump Info:
   Message: Success
```

Adapter Port Beacon Configuration

The `port beacon` command configures beacon values of the selected Marvell Adapter port:

~# esxcli qlfc qcc port beacon

Syntax

```
esxcli qlfc qcc port beacon {cmd} [cmd options]
```

Available Commands

`get`
Shows beacon configuration of the specified Marvell Adapter port. See Get Port Beacon Configuration.

`set`
Sets the beacon configuration of the specified Marvell Adapter port to flash on the adapter. See Set Port Beacon Configuration.

Get Port Beacon Configuration

The `port beacon get` command shows port-level beacon configuration for the specified Marvell Adapter port. This command is supported only on the physical port (not on the NPIV virtual port).

```
~# esxcli qlfc qcc port beacon get
```

Syntax

```
esxcli qlfc qcc port beacon get [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port beacon get` command on a local host:

```
~# esxcli qlfc qcc port beacon get -i FC_51:40:2e:c0:12:3c:f4:a0
```

The following example shows the `port beacon get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port beacon
get -i FC_51:40:2e:c0:12:3c:f4:a0
```

The output for both commands is:

```
Beacon Info:
   Beacon State: OFF
   Message: Success
```

Set Port Beacon Configuration

The `port beacon set` command sets port-level beacon configuration for the specified Marvell Adapter port. This command is supported only on the physical port (not on the NPIV virtual port).

```
~# esxcli qlfc qcc port beacon set
```

Syntax

```
esxcli qlfc qcc port beacon set [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

`-b|--beacon-state <str>` = Beacon state

Parameter	Value (strings are case-sensitive)
`Identifier`	Port identifier from the `port list` command
`beacon-state`	ON, OFF

Example

The following example shows the `port beacon set` command on a local host:

```
~# esxcli qlfc qcc port beacon set -i FC_51:40:2e:c0:12:3c:f4:a0
-b ON
```

The following example shows the `port beacon set` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port beacon
set -i FC_51:40:2e:c0:12:3c:f4:a0 -b ON
```

The output for both commands is:

```
Beacon Info:
   Beacon State: ON
   Message: Success
```

Adapter Port Diagnostics (DPort) Configuration

The `port dportdiag` command sets the port in diagnostic port (DPort) mode and runs DPort diagnostic commands on the selected Marvell Adapter port:

```
~# esxcli qlfc qcc port dportdiag
```

Syntax

```
esxcli qlfc qcc port dportdiag {cmd} [cmd options]
```

Available Commands

`get`
Shows DPort configuration of the specified Marvell Adapter port. See Get DPort Configuration.

`set`
Sets the DPort configuration of the specified Marvell Adapter port. See Set DPort Configuration.

`run`
Runs a DPort operation on the specified Marvell Adapter port. See Run DPort Diagnostics.

Get DPort Configuration

The `port dportdiag get` command shows the port-level DPort configuration for the specified Marvell Adapter port. This command is supported only on the physical port (not on the NPIV virtual port).

```
~# esxcli qlfc qcc port dportdiag get
```

Syntax

```
esxcli qlfc qcc port dportdiag get [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port dportdiag get` command on a local host:

```
~# esxcli qlfc qcc port dportdiag get -i
FC_51:40:2e:c0:12:3c:f4:a0
```

The following example shows the `port dportdiag get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
dportdiag get -i FC_51-40:2e:c0:12:3c:f4:a0
```

The output for both commands is:

```
DPort Diag Info:
   Diagnostics Port Status: false
   Message: Success
```

Set DPort Configuration

The `port dportdiag set` command sets port-level DPort configuration for the specified Marvell Adapter port. This command is supported only on the physical port (not) on NPIV virtual port.

```
~# esxcli qlfc qcc port dportdiag set
```

Syntax

```
esxcli qlfc qcc port dportdiag set [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

`-m|--mode <str>` = DPort mode

Parameter	Value (strings are case-sensitive)
`Identifier`	Port identifier from the `port list` command
`mode`	true, false

Example

The following example shows the `port dportdiag set` command on a local host:

```
~# esxcli qlfc qcc port dportdiag set -i
FC_51:40:2e:c0:12:3c:f4:a0 -m true
```

The following example shows the `port dportdiag set` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
dportdiag set -i FC_51:40:2e:c0:12:3c:f4:a0 -m true
```

The output for both commands is:

```
DPort Diag Info:
   Diagnostics Port Status: true
   Message: Success
   Reboot Required: true
```

Run DPort Diagnostics

The `port dportdiag run` command runs a host port-level DPort configuration for the specified Marvell Adapter port. This command is supported only on the physical port (not on the NPIV virtual port).

```
~# esxcli qlfc qcc port dportdiag run
```

Syntax

```
esxcli qlfc qcc port dportdiag run [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port dportdiag run` command on a local host:

```
~# esxcli qlfc qcc port dportdiag run -i
FC_51:40:2e:c0:12:3c:f4:a0
```

The following example shows the `port dportdiag get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
dportdiag run -i FC_51:40:2e:c0:12:3c:f4:a0
```

The output for both commands is:

```
DPort Diag Info:
   DPort Diag Result:
         Data:
      Start Time: Tue Nov 30 22:23:34 2021
      Stop Time: Tue Nov 30 22:23:53 2021
      HBA Port Value: 0x1
      HBA Port Status: 1
      HBA Port Result: Passed
      Electrical Loopback Value: 0xd2
      Electrical Loopback Status: 1
      Electrical Loopback Result: Passed
      Optical Loopback Value: 0xd3
      Optical Loopback Status: 1
      Optical Loopback Result: Passed
      Link Traffic Value: 0xd5
      Link Traffic Status: 2
      Link Traffic Result: Skipped
      Tx Power: 0x173a
      Rx Power: 0x184b
      Mbx1: 0x41
      Mbx2: 0xe000
      Roundtrip Link Latency: 176 ns
      Cable Link Distance: 3 meters
      Allowable Power Loss: 0.014000
```

```
        Egress Power: Tx: -2.257750 dBm, Rx: -2.123978 dBm,
                      Diff: -0.133772 dBm
        Ingress Power: Tx: -1.930742 dBm, Rx: -2.062794 dBm,
                       Diff: 0.132052 dBm

    Message: Success
```

Adapter Port Common Pass-thru (CT) Diagnostics

NOTE

This command is supported for FCP targets and NVMe-FC targets.

The `port ctdiag` command runs common pass-thru (CT) diagnostic commands on the selected Marvell Adapter port:

```
~# esxcli qlfc qcc port ctdiag
```

Syntax

```
esxcli qlfc qcc port ctdiag {cmd} [cmd options]
```

Available Commands

`run`
Runs CT diagnostics on the specified Marvell Adapter port. See Run Port CT Diagnostics.

Run Port CT Diagnostics

The `port ctdiag run` command runs host port-level CT diagnostics commands for the specified port on the Marvell Adapter. This command is supported on the physical port, but not on the NPIV virtual port.

```
~# esxcli qlfc qcc port ctdiag run
```

Syntax

```
esxcli qlfc qcc port ctdiag run [cmd options]
```

Keywords

`-i|--identifier <str>`	=	Port identifier (required). Obtain the identifier from the port `list` output.
`-n|--number-of-tests <long>`	=	Number of test iterations
`-e|--stop-on-error <str>`	=	Test stop on error

```
-t|--target <str>  =  Target WWPN (":" separated)
-p|--test-type <str>  =  Test type
```

Parameter	Value (strings are case-sensitive)
identifier	Port identifier from the port list command
number-of-tests	1–65535
stop-on-error	true, false
target	":" separated target WWPN
test-type	CTPing, CTFCTraceRoute

Example

The following example shows the port ctdiag run command on a local host:

```
~# esxcli qlfc qcc port ctdiag run -i FC_51:40:2e:c0:12:3c:f4:a0
-n 2 -e true -t 50:0a:09:82:90:a2:ce:4d -p CTPing
```

The following example shows the port ctdiag run command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port ctdiag
run -i FC_51:40:2e:c0:12:3c:f4:a0 -n 2 -e true -t
50:0a:09:82:90:a2:ce:4d -p CTPing
```

The output for both commands is:

```
CT Diag Result:
   Diagnostic Test: CTPing
   Test Result: Success
```

Adapter Port Echo ELS Diagnostics

NOTE

This command is supported for FCP targets and NVMe-FC targets.

The port fcecho command runs Fibre Channel Echo ELS diagnostic commands on the specified Marvell Adapter port:

```
~# esxcli qlfc qcc port fcecho
```

Syntax

```
esxcli qlfc qcc port fcecho {cmd} [cmd options]
```

Available Commands

`run`
Runs Fibre Channel Echo ELS diagnostics on the specified Marvell Adapter port. See “Run Port Fibre Channel Echo ELS Diagnostics” on page 207.

Run Port Fibre Channel Echo ELS Diagnostics

The `port fcecho run` command runs host port-level Fibre Channel Echo ELS diagnostics commands for the specified Marvell Adapter port. This command is supported on the physical port, but not on the NPIV virtual port.

```
~# esxcli qlfc qcc port fcecho run
```

Syntax

```
esxcli qlfc qcc port fcecho run [cmd options]
```

Keywords

`-i|--identifier <str>`	=	Port identifier (required). Obtain the identifier from the port `list` output.
`-n|--number-of-tests <long>`	=	Number of test iterations
`-e|--stop-on-error <str>`	=	Test stop on error
`-t|--target <str>`	=	Target WWPN (“**:**” separated)

Parameter	Value (strings are case-sensitive)
`identifier`	Port identifier from the port `list` command
`number-of-tests`	1–65535
`stop-on-error`	true, false
`target`	“**:**” separated target WWPN

Example

The following example shows the `port fcecho run` command on a local host:

```
~# esxcli qlfc qcc port fcecho run -i FC_51:40:2e:c0:12:3c:f4:a0
-n 2 -e true -t 50:0a:09:82:90:a2:ce:4d
```

The following example shows the `port fcecho run` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port fcecho
run -i FC_51:40:2e:c0:12:3c:f4:a0 -n 2 -e true -t
50:0a:09:82:90:a2:ce:4d
```

The output for both commands is:

```
FC Echo ELS Result:
   FCEcho Test Results:
         Destination WWN: 50:0a:09:82:90:a2:ce:4d
         Echo Status: Success
         Frame Received: 1
         Frame Sent: 1
         Ping Sequence Number: 1
         Response Length: 256
         Response Time:   0.00 (ms)

         Destination WWN: 50:0a:09:82:90:a2:ce:4d
         Echo Status: Success
         Frame Received: 1
         Frame Sent: 1
         Ping Sequence Number: 2
         Response Length: 256
         Response Time:   0.00 (ms)
         Message: Success
```

Adapter Port Ping Diagnostics

NOTE

This command is supported for FCP targets and NVMe-FC targets.

The `port fcping` command runs Fibre Channel ping diagnostic commands on the selected Marvell Adapter port:

```
~# esxcli qlfc qcc port fcping
```

Syntax

```
esxcli qlfc qcc port fcping {cmd} [cmd options]
```

Available Commands

`run`
Runs Fibre Channel ping diagnostics on the specified Marvell Adapter port. See Run Port Fibre Channel Ping Diagnostics.

Run Port Fibre Channel Ping Diagnostics

The `port fcping run` command runs host port-level Fibre Channel ping diagnostics commands for the specified Marvell Adapter port. This command is supported on physical port, but not on the NPIV virtual port.

```
~# esxcli qlfc qcc port fcping run
```

Syntax

```
esxcli qlfc qcc port fcping run [cmd options]
```

Keywords

`-i|--identifier <str>`	=	Port identifier (required). Obtain the identifier from the port `list` output.
`-n|--number-of-tests <long>`	=	Number of test iterations
`-e|--stop-on-error <str>`	=	Test stop on error
`-t|--target <str>`	=	Target WWPN (“:” separated)

Parameter	Value (strings are case-sensitive)
`identifier`	Port identifier from the port `list` command
`number-of-tests`	1–65535
`stop-on-error`	true, false
`target`	“:” separated target WWPN

Example

The following example shows the `port fcping run` command on a local host:

```
~# esxcli qlfc qcc port fcping run -i FC_51:40:2e:c0:12:3c:f4:a0
-n 2 -e true -t 50:0a:09:82:90:a2:ce:4d
```

The following example shows the `port fcping run` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port fcping
run -i FC_51:40:2e:c0:12:3c:f4:a0 -n 2 -e true -t
50:0a:09:82:90:a2:ce:4d
```

The output for both commands is:

```
FC Ping Result:
   Diagnostic Test: FCPing
   Test Result: Success
```

Adapter Port Read Port Diagnostics (RDP) Information

The `port rdp` command shows port diagnostics information about the selected Marvell Adapter port:

```
~# esxcli qlfc qcc port rdp
```

Syntax

```
esxcli qlfc qcc port rdp {cmd} [cmd options]
```

Available Commands

`get`
Shows information about the specified Marvell Adapter port. See Get Read Port Diagnostics Information.

Get Read Port Diagnostics Information

The `port rdp get` command shows port-level diagnostics information for the specified Marvell Adapter port.

```
~# esxcli qlfc qcc port rdp get
```

Syntax

```
esxcli qlfc qcc port rdp get [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port rdp get` command on a local host:

```
~# esxcli qlfc qcc port rdp get -i FC_51:40:2e:c0:12:3c:f4:a0
```

The following example shows the `port rdp get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port rdp get
-i FC_51:40:2e:c0:12:3c:f4:a0
```

The output for both commands is:

```
RDP Information:
   Message: Success
   Read Diagnostics Parameters Data:
         RDPInfo:
```

```
--------------------------------
Diagnostics Parameters Information
--------------------------------
Descriptor Tag: Link Service Request Information
Descriptor Len: 4 Bytes
Descriptor Value: 0x18000000

--------------------------------
Port Speed Information
--------------------------------
Port Speed Capabilities: 32 16 8 4 Gbps
Port Operating Speed: 32 Gbps

--------------------------------
Link Error Status Block Information
--------------------------------
Link Failure Count: 13
Loss Of Sync Count: 0
Loss Of Signal Count: 13
Primary Sequence Error Count: 0
Invalid Transmit Word Count: 0
Invalid CRC Count: 0
PN Port Physical Type: 0x40000000
        The sending VN_Port uses an FC-FS-3 PN_Port or PF Port

--------------------------------
Port Name Information
--------------------------------
Node WWN: 10:00:88:94:71:97:67:63
Port WWN: 20:11:88:94:71:97:67:63

--------------------------------
Port Name Information
--------------------------------
Node WWN: 51:40:2e:c0:12:3c:f4:a1
Port WWN: 51:40:2e:c0:12:3c:f4:a0

--------------------------------
SFP Diagnostics Param Information
```

```
---------------------------------
Temperature: 58 (C)
Vcc: 3.338599920 V
Tx Bias: 7.642000198 mA
Tx Power: 0.737200022 mW
Rx Power: 0.781500041 mW
SFP Flag: 0x0051
        Port Tx Type: Short Wave Laser
        Connector Type: SFP+
        Optical Port: On
        SFP Diagnostics Parameters not valid: Off
        FEC Active: Off

---------------------------------
FEC Status Information
---------------------------------
Correctable Blocks: 0
UnCorrectable Blocks: 58

---------------------------------
Buffer Credits Status Information
---------------------------------
FC Port Buffer To Buffer Credits: 20
Attached FC Port Buffer To Buffer Credits: 48
Nominal FC Port RTT: 0 ns

---------------------------------
Optical Product Data Information
---------------------------------
Vendor Name: BROCADE
Part Number: 57-1000333-01
Serial Number: JAF418500000BL0
Revision:
Date: 181211

---------------------------------
Optical Element Data Information
---------------------------------
Temperature High Alarm: 75.00
```

```
Temperature Low Alarm: -5.00
Temperature High Warning: 70.00
Temperature Low Warning: 0.00

--------------------------------
Optical Element Data Information
--------------------------------
Voltage High Alarm: 3.599999905
Voltage Low Alarm: 3.000000000
Voltage High Warning: 3.500000000
Voltage Low Warning: 3.099999905

--------------------------------
Optical Element Data Information
--------------------------------
Tx Bias High Alarm: 12.000000954
Tx Bias Low Alarm: 1.000000000
Tx Bias High Warning: 11.500000954
Tx Bias Low Warning: 2.000000000

--------------------------------
Optical Element Data Information
--------------------------------
Tx Power High Alarm: 1.995300174
Tx Power Low Alarm: 0.125900000
Tx Power High Warning: 1.584900141
Tx Power Low Warning: 0.158500001

--------------------------------
Optical Element Data Information
--------------------------------
Rx Power High Alarm: 1.995300174
Rx Power Low Alarm: 0.010000001
Rx Power High Warning: 1.584900141
Rx Power Low Warning: 0.015800001
```

Adapter Port Loopback Diagnostics

The `port loopback` command runs loopback diagnostic commands on the specified Marvell Adapter port:

`~# esxcli qlfc qcc port loopback`

Syntax

```
esxcli qlfc qcc port loopback {cmd} [cmd options]
```

Available Commands

`run`
Runs loopback diagnostics on the specified Marvell Adapter port. See Run Port Loopback Diagnostics.

Run Port Loopback Diagnostics

The `port loopback run` command runs host port-level loopback diagnostics commands for the specified Marvell Adapter port. This command is supported on the physical port, but not on the NPIV virtual port.

`~# esxcli qlfc qcc port loopback run`

Syntax

```
esxcli qlfc qcc port loopback run [cmd options]
```

Keywords

`-i|--identifier <str>`	=	Port identifier (required). Obtain the identifier from the port `list` output.
`-n|--number-of-tests <long>`	=	Number of test iterations
`-t|--test-increment <long>`	=	Test increment count
`-e|--stop-on-error <str>`	=	Test stop on error
`-l|--loopback-type <str>`	=	Loopback test type
`-z|--data-size <str>`	=	Data size
`-p|--data-pattern-type <str>`	=	Loopback buffer data pattern type
`-d|--data-pattern <str>`	=	Custom 8-byte data pattern for loopback buffer

Parameter	Value (strings are case-sensitive)
identifier	Port identifier from the port `list` command
number-of-tests	1–65535
test-increment	1–65535
stop-on-error	Ignore, Stop, Loop
loopback-type	Internal_10bit, Internal, External, FCoE_Echo (only for 8400 and legacy 8100/8200/8300 Series Adapters)
data-size	8, 16, 32, 64, 128, 256, 512, 1024, 2048, 4096, 8192, 16384, 32768, 65535
data-pattern-type	Random, CRPAT, CJTPAT, CSPAT, 00000000, 01010101, 01011010, 10100101, 10101010, 11111111, Custom
data-pattern	8-byte used defined data pattern (only for Custom `data-pattern-type`)

Example

The following example shows the `port loopback run` command on a local host:

```
~# esxcli qlfc qcc port loopback run -i FC_51:40:2e:c0:12:3c:f4:a0
-n 2 -t 5 -e Stop -l External -z 512 -p Random
```

The following example shows the `port loopback run` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port loopback
run -i FC_51:40:2e:c0:12:3c:f4:a0 -n 2 -t 5 -e Stop -l External -z
512 -p Random
```

The output for both commands is:

```
Loopback Test Information:
   Loopback Test Result:
         Crc Error: 0
         Disparity Error: 0
         Failure Count: 0
         Frame Length Error: 0
         Success Count: 5
         Test Iteration: 5
         Test Status: Success
   Message: Success
```

Adapter Port Read and Write Buffer Diagnostics

The `port rwbuffer` command runs read and write buffer diagnostic commands on the specified Marvell Adapter port:

```
~# esxcli qlfc qcc port rwbuffer
```

Syntax

```
esxcli qlfc qcc port rwbuffer {cmd} [cmd options]
```

Available Commands

`run`
Runs read and write buffer diagnostics on the specified Marvell Adapter port. See Run Port Read and Write Buffer Diagnostics.

Run Port Read and Write Buffer Diagnostics

The `port rwbuffer run` command runs host port-level read-write buffer diagnostics commands for the specified Marvell Adapter port. This command is supported on the physical port, but not on the NPIV virtual port.

```
~# esxcli qlfc qcc port rwbuffer run
```

Syntax

```
esxcli qlfc qcc port rwbuffer run [cmd options]
```

Keywords

`-i|--identifier <str>`	=	Port identifier (required). Obtain the identifier from the port `list` output.
`-n|--number-of-tests <long>`	=	Number of test iterations
`-t|--test-increment <long>`	=	Test increment count
`-e|--stop-on-error <str>`	=	Stop on error
`-z|--data-size <str>`	=	Data size
`-p|--data-pattern-type <str>`	=	Loopback buffer data pattern type
`-d|--data-pattern <str>`	=	User defined 8-byte data pattern for the loopback buffer. Use only when `data-pattern-type` parameter is Custom.

Parameter	Value (strings are case-sensitive)
identifier	Port identifier from the port `list` command
number-of-tests	1–10000
test-increment	1–10000
stop-on-error	Ignore, Stop, Loop
data-size	8, 16, 32, 64, 128
data-pattern-type	Random, CRPAT, CJTPAT, CSPAT, 00000000, 01010101, 01011010, 10100101, 10101010, 11111111, Custom
data-pattern	8-byte used defined data pattern (only for Custom `data-pattern-type`)

Example

The following example shows the `port rwbuffer run` command on a local host:

```
~# esxcli qlfc qcc port rwbuffer run -i FC_51:40:2e:c0:12:3c:f4:a0
-n 2 -t 5 -e Stop -z 64 -p Random
```

The following example shows the `port rwbuffer run` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port rwbuffer
run -i FC_51:40:2e:c0:12:3c:f4:a0 -n 2 -t 5 -e Stop -z 64 -p Random
```

The output for both commands is:

```
Read Write Buffer Test Information:
   Message: Success
   RWBuffer Test Result:
         Data Miscompare: 0
         Failure Count: 0
         Invalid CRC: 0
         Link Failure: 0
         Loss Of Signal: 0
         Loss Of Sync: 0
         Port ID: 0a:00:00
         Success Count: 5
         Test Iteration: 5
         Test Status: Success
```

Adapter Port Universal SAN Congestion Mitigation Status

NOTE

SAN congestion management (SCM) is a common noun, and describes a standards-based Fibre Channel technology.

Universal SAN Congestion Mitigation (USCM) is Marvell's IP, and describes Marvell's capabilities that encompass SCM and additional functionalities to further assist users.

USCM is supported only on 2800 Series Adapters, 2770 Series Adapters, and 2690 Series Adapters.

The `port scmchk` command displays the USCM status of the specified Marvell Adapter port:

```
~# esxcli qlfc qcc port scmchk
```

Syntax

```
esxcli qlfc qcc port scmchk {cmd} [cmd options]
```

Available Commands

`get`
Shows USCM status information of the specified Marvell Adapter port. See Get Initiator Port USCM Status.

Get Initiator Port USCM Status

The `port scmchk get` command shows port-level USCM status for the specified Marvell Adapter initiator port:

```
~# esxcli qlfc qcc port scmchk get
```

Syntax

```
esxcli qlfc qcc port scmchk get [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port scmchk get` command on a local host:

```
~# esxcli qlfc qcc port scmchk get -i FC_51:40:2e:c0:12:3c:f4:a0
```

The following example shows the `port scmchk get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port scmchk
get -i FC_51:40:2e:c0:12:3c:f4:a0
```

The output for both commands is:

```
Port Congestion Information:
   Message: Success
   Port Congestion Status:
         Data:
      Node Name: 51:40:2e:c0:12:3c:f4:a1
      Port Name: 51:40:2e:c0:12:3c:f4:a0
      Port Id: 01:02:00
      Port State: Online
      Congestion Current State: Healthy
      Congestion Severity: None
      Link Integrity Events: No
      Delivery Notification Events: No
      Seconds Since Last Event: 409
      Fabric Connection Flags: Rdf Completed
      Virtual Lane: Operational
      I/O Throttling: None
```

Not all of the port parameters in this section are available on all systems. The attributes are based on the adapter driver version.

Initiator Port Status

The USCM initiator port congestion status (Table 4-1) indicates the current status of the particular adapter port, based on congestion events (both extended link service (ELS) and signals) from the switch. At a given point in time, the adapter port is either *congested* or *healthy*. The other status indicates the severity of congestion and the time since the last congestion event.

Contact the fabric/switch vendor for more information relating to congestion severity within their products.

Table 4-1. USCM Initiator Port Status

Status	Description
Congestion Current State	**Healthy** **Congested**

Table 4-1. USCM Initiator Port Status (Continued)

Status	Description
Congestion Severity	**Warning**. Congestion is building and may have reached a severe level. **Alarm**. Congestion has reached a severe level. **None**. No congestion present. **Reserved** ***NOTE:*** For more FPIN information, see the user's guide for your switch.
Link Integrity Events	**Yes.** A link event has been received. **No**
Delivery Notification Events	**Yes.** A delivery notification event has been received. **No**
Seconds Since Last Event	Event time period (in seconds) since the last congestion notification was received.
Fabric Connection Flags	**RDF Rejected.** The adapter is either not connected or does not support RDF; or the connected switch does not support USCM. **RDF Completed.** Either the connected Brocade or Cisco switch does not support virtual lanes or the virtual lane feature is disabled at the initiator port. **RDF Completed (Cisco).** The connected Cisco switch has virtual lanes up and running. ***NOTE:*** RDF stands for Registration Diagnostic Function.
Virtual Lane [a]	**Operational**. Virtual lane negotiation is complete and fully functional. **Non-operational** Virtual lane negotiation with the switch failed. **Disabled** Virtual lanes are disabled at either the driver or the initiator port parameter (NVRAM).

Table 4-1. USCM Initiator Port Status (Continued)

Status	Description
`I/O Throttling` [a]	**`Active`** I/O throttling is currently active. The I/O queue depth is ramping up or down. **`None`** I/O throttling is not active. I/Os are running with the normal queue depth value. **`Disabled`** I/O throttling is disabled on the adapter. **`NA`** I/O throttling is not supported on the adapter.

[a] For more information about virtual lanes and I/O throttling, see the appropriate Marvell QLogic Fibre Channel Adapter user's guide.

Adapter Port Universal SAN Congestion Mitigation Statistics

For a description of USCM, see the following NOTE.

The `port scmstats` command displays the USCM statistics of the selected FC port:

```
~# esxcli qlfc qcc port scmstats
```

Syntax

```
esxcli qlfc qcc port scmstats {cmd} [cmd options]
```

Available Commands

`get`
Shows USCM statistics information of the specified Marvell Adapter Fibre Channel initiator port. See Get Initiator Port USCM Statistics.

`clear`
Clears USCM statistics of the specified Marvell Adapter Fibre Channel initiator port and all the target ports connected to that initiator. See Clear Initiator Port and Connected Targets USCM Statistics.

Get Initiator Port USCM Statistics

USCM statistics are gathered for all ports on the Marvell QLogic FC adapter and connected targets in the configured zones with active sessions.

- USCM statistics are not gathered for other initiators in the configured zones.
- USCM statistics track the following types of Fabric Performance Impact Notification (FPIN) Extended Link Services (ELSs) to provide SAN congestion awareness:

FPIN ELS Statistic	Initiator Port	Target Port
Congestion	✓	—
Peer congestion	—	✓
Link integrity	✓	✓
Delivery	✓	✓

- USCM also displays a set of congestion mitigation statistics that reflect actions taken by the adapter to minimize the impact of congestion caused by the endpoints.

You can view the USCM congestion mitigation status and statistics; the statistics can also be reset.

For target USCM statistics, see “Adapter Target Universal SAN Congestion Mitigation Statistics” on page 236.

The `port scmstats get` command shows port-level USCM statistics for the specified Marvell Adapter port:

```
~# esxcli qlfc qcc port scmstats get
```

Syntax

```
esxcli qlfc qcc port scmstats get [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port scmstats get` command on a local host:

```
~# esxcli qlfc qcc port scmstats get -i FC_51:40:2e:c0:12:3c:f4:a0
```

The following example shows the `port scmstats get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
scsmstats get -i FC_51:40:2e:c0:12:3c:f4:a0
```

The output for both commands is:

```
Port SCM Statistics:
   Message: Success
   Port SCMStats:
         Data:
      Node Name: 51:40:2e:c0:12:3c:f4:a1
      Port Name: 51:40:2e:c0:12:3c:f4:a0
      Port Id: 0a:11:00
      Port State: Online
      ----------------------------------
      Congention Mitigation
      ----------------------------------
      Congestion Alarm Count: 3
      Congestion Warning Count: 8
      Cleared Congestion Count: 21
      Throttled Up Count: 2
      Throttled Down Count: 3
      Bottomed Out Count: 5
      Returned Busy Count:
      ----------------------------------
      Rx Fabric Performance Impact
      Notifications
      ----------------------------------
      Link Failure Count: 0
      Link Unknown Event Count: 0
      Loss Of Sync Count: 0
      Loss Of Signal Count: 0
      Link Device Specific Event Count: 0
      Primitive Sequence Protocol Error Count: 0
      Invalid Transmission Word Count: 0
```

```
Invalid CRC Count: 0
Link Uncorrectable FEC Count: 0
Delivery Failure Unknown Count: 0
Delivery Timeout Count: 0
Delivery Unable To Route Count: 0
Delivery Failure Device Specific Count: 0
Congestion Clear Count: 0
Congestion Lost Credit Count: 0
Congestion Credit Stall Count: 0
Congestion Oversubscription Count: 0
Congestion Device Specific Count: 0
```

Not all of the port parameters in this section are available on all systems. The attributes are based on the adapter driver version.

Initiator Port Statistics

The USCM initiator port statistics (Table 4-2) indicate how many times a congestion event has occurred since the counters were reset.

Table 4-2. USCM Initiator Port Congestion Mitigation Statistics

Statistic	Description
`Congestion Alarm Count`	Counter for the number of alarm events in Congestion Severity.
`Congestion Warning Count`	Counter for the number of warning events in Congestion Severity.
`Cleared Congestion Count`	The number of times the congestion event was cleared for this port.
`Throttled Down Count`	The fabric is congested at this port. ■ Each increment indicates that the port is receiving congestion notifications from the fabric. ■ For each increment, the port decreases the I/O bandwidth. ■ Incrementing stops when the low watermark performance for the port is reached.

Table 4-2. USCM Initiator Port Congestion Mitigation Statistics (Continued)

Statistic	Description
`Throttled Up Count`	The fabric is *not* congested at this port. ■ Each increment indicates that the port is *not* receiving congestion notifications from the fabric. ■ For each increment, the port increases the I/O bandwidth (up to the maximum I/O bandwidth for the port). ■ Incrementing starts only after the Throttle Down Count increments (and congestion notifications abate). ■ Incrementing stops when the port reaches its optimal performance.
`Bottomed Out Count`	This port is operating at its low watermark performance and the fabric continues to be congested at this port. ■ Each increment indicates that the port is receiving congestion notifications from the fabric. ■ This port cannot decrease the I/O bandwidth any further.
`Returned Busy Count`	The number of times the driver returns an I/O request queued by the initiator's storage stack. This request must be retried because the port is congested. The count is incremented faster when the driver is throttling down, and slower when the driver is throttling up.

Clear Initiator Port and Connected Targets USCM Statistics

The `port scmstats clear` command clears port-level USCM statistics for both the specified Marvell Adapter Fibre Channel initiator port and all the targets connected to that port:

```
~# esxcli qlfc qcc port scmstats get
```

Syntax

```
esxcli qlfc qcc port scmstats clear [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port scmstats clear` command on a local host:

```
~# esxcli qlfc qcc port scmstats clear -i
FC_51:40:2e:c0:12:3c:f4:a0
```

The following example shows the `port scmstats clear` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port scmstats
clear -i FC_51:40:2e:c0:12:3c:f4:a0
```

The output for both commands is:

```
SCM Statistics Clear:
   Message: Success
```

Adapter Port Universal SAN Congestion Mitigation Profile

The `port scmprofile` command displays the USCM congestion profile of the selected FC port:

```
~# esxcli qlfc qcc port scmprofile
```

Syntax

```
esxcli qlfc qcc port scmprofile {cmd} [cmd options]
```

Available Commands

`get`
Shows USCM congestion profile information of the specified Marvell Adapter Fibre Channel port. See Get Port Universal SAN Congestion Mitigation Profile.

`set`
Sets USCM congestion profile information of the specified Marvell Adapter port. See Set Port Universal SAN Congestion Mitigation Profile.

For more information about USCM profiles, see the appropriate Marvell QLogic Fibre Channel Adapter user's guide.

Get Port Universal SAN Congestion Mitigation Profile

The `port scmprofile get` command displays the port-level USCM profile settings of the selected FC port:

```
~# esxcli qlfc qcc port scmprofile get
```

Syntax

```
esxcli qlfc qcc port scmprofile get [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

Example

The following example shows the `port scmprofile get` command on a local host:

```
~# esxcli qlfc qcc port scmprofile get -i
FC_51:40:2e:c0:12:3c:f4:a0
```

The following example shows the `port scmprofile get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
scmprofile get -i FC_51:40:2e:c0:12:3c:f4:a0
```

The output for both commands is:

```
Port USCM Profile:
   Congestion Profile:
         Data:
      Node Name: 51:40:2e:c0:12:3c:f4:a1
      Port Name: 51:40:2e:c0:12:3c:f4:a0
      Port Id: 0a:11:00
      Port State: Online
      USCM Configuration Status: true
      USCM Profile Management: Driver_Settings
      USCM Profile Activation: Monitor_Only

   Message: Success
```

Set Port Universal SAN Congestion Mitigation Profile

The `port scmprofile set` command sets the port-level USCM profile settings of the selected FC port:

```
~# esxcli qlfc qcc port scmprofile set
```

Syntax

```
esxcli qlfc qcc port scmprofile set [cmd options]
```

Keywords

`-i|--identifier <str>` = Port identifier (required). Obtain the identifier from the port `list` output.

`-a|--activation <str>` = Congestion profile activation value

`-r|--restore <str>` = Restore congestion profile to the default driver setting

Parameter	Value (strings are case-sensitive)
`identifier`	Port identifier from the port `list` command
`activation`	`Monitor_Only` (default), `Conservative`, `Moderate`, `Aggressive` These values are referred to as the *port profiles*; see Table 4-3.
`restore`	`Driver_Settings`

The `activation` and `restore` parameters are mutually exclusive.

Settings are activated immediately without reboot.

Settings are persistent across reboots.

The `restore` parameter sets profile settings defined using either the driver module parameter or the default value (if not defined using the driver module parameter). The default value is Monitor_Only.

Table 4-3. USCM Adapter Port Profiles

Profile	Description
USCM Configuration Status	Indicates if the USCM feature of the specified adapter port is supported. Valid values are: **True** **False**
USCM Profile Management	Indicates the current active profile setting for congestion on the specified adapter port. Valid values are: **Driver_Settings** (default). The USCM profile is set using either a Windows driver registry parameter; or a Linux or VMware ESXi driver module parameter. **NVRAM Settings**. The USCM profile is set in the adapter NVRAM.

Table 4-3. USCM Adapter Port Profiles (Continued)

Profile	Description
USCM Profile Activation	Valid values are: **Monitor Only** (default). Records adapter performance and congestion history for review. No actions are taken to resolve congestion. **Conservative.** Maintains optimum throughput while gradually reducing congestion. Queue depth (outstanding I/Os) is reduced to half of the current value as part of the throttle down operation. Marvell recommends this setting for high-priority workloads. **Moderate.** Queue depth (outstanding I/Os) is reduced to one-quarter of the current value as part of the throttle down operation. **Aggressive.** Reduces congestion on priority while reducing throughput. Queue depth (outstanding I/Os) is reduced to one-eighth of the current value as part of the throttle down operation. Marvell recommends this setting for low-priority workloads.

Example

The following example shows the `port scmprofile set` activation command on a local host:

```
~# esxcli qlfc qcc port scmprofile set -i
FC_51:40:2e:c0:12:3c:f4:a0 -a Aggressive
```

The following example shows the `port scmprofile set` activation command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
scmprofile set -i FC_51:40:2e:c0:12:3c:f4:a0 -a Aggressive
```

The output for both commands is:

```
Port USCM Profile:
   Congestion Profile:
         Data:
      Node Name: 51:40:2e:c0:12:3c:f4:a1
```

```
      Port Name: 51:40:2e:c0:12:3c:f4:a0
      Port Id: 0a:11:00
      Port State: Online
      USCM Configuration Status: true
      USCM Profile Management: NVRAM
      USCM Profile Activation: Aggressive

   Message: Success
```

The following example shows the `port scmprofile set` restore command on a local host:

```
~# esxcli qlfc qcc port scmprofile set -i
FC_51:40:2e:c0:12:3c:f4:a0 -r Driver_Settings
```

The following example shows the `port scmprofile set` restore command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
scmprofile set -i FC_51:40:2e:c0:12:3c:f4:a0 -r Driver_Settings
```

The output for both commands is:

```
Port USCM Profile:
   Congestion Profile:
         Data:
      Node Name: 51:40:2e:c0:12:3c:f4:a1
      Port Name: 51:40:2e:c0:12:3c:f4:a0
      Port Id: 0a:11:00
      Port State: Online
      USCM Configuration Status: true
      USCM Profile Management: Driver_Settings
      USCM Profile Activation: Monitor_Only

   Message: Success
```

Adapter Target Namespace Commands and Sub-namespaces

This section provides information on the following:

- Adapter Target List
- Adapter Target Information
- Adapter Target Universal SAN Congestion Mitigation Status

- Adapter Target Universal SAN Congestion Mitigation Statistics
- Adapter Target iiDMA Configuration

NOTE

These commands are supported for FCP targets and NVMe-FC targets.

To view all Marvell Adapter target namespace and subspace commands, issue the following:

```
# esxcli qlfc qcc target
```

Syntax

```
esxcli qlfc qcc target {cmd} [cmd options]
```

Namespaces

Available namespaces include the following:

Namespace	Description
`info`	QCC CLI for target information
`scmchk`	QCC CLI for target USCM status
`scmstats`	QCC CLI for target USCM statistics
`iidma`	QCC CLI for target maximum link speed
`stats`	QCC CLI for target link statistics

Available Commands

`list`
Lists identifiers for the Marvell Adapter targets discovered on the host. See Adapter Target List.

Adapter Target List

The `target list` command shows all the targets detected, as follows:

```
~# esxcli qlfc qcc target list
```

The following example shows the `target list` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc target list
```

The command output is:

```
Targets
-------------------------------------------------------
FC_51:40:2e:c0:12:3c:f4:a2_Target_50:0a:09:81:90:a2:ce:4d
```

```
FC_51:40:2e:c0:12:3c:f4:a2_Target_ 20:02:00:11:0d:27:b2:00
```

Adapter Target Information

The `target info` command shows target information about the selected target:

`~# esxcli qlfc qcc target info`

Syntax

`esxcli qlfc qcc target info {cmd} [cmd options]`

Available Commands

`get`
Shows information about the specified target on the Marvell Adapter port. See Get Target Information.

Get Target Information

The `target info get` command shows target information for the specified target.

`~# esxcli qlfc qcc target info get`

Syntax

`esxcli qlfc qcc target info get [cmd options]`

Keywords

`-i|--identifier <str>` = Target identifier (required). Obtain the identifier from the target `list` output.

Example

The following example shows the `target info get` command for FCP targets on a local host:

`~# esxcli qlfc qcc target info get -i FC_51:40:2e:c0:12:3c:f4:a2_Target_50:0a:09:81:90:a2:ce:4d`

The following example shows the `target info get` command for FCP targets on a remote host:

`# esxcli -s 172.27.9.144 -u root -p password qlfc qcc target info get -i FC_51:40:2e:c0:12:3c:f4:a2_Target_50:0a:09:81:90:a2:ce:4d`

The output for both commands is:

```
Target Information:
   Message: Success
   Target Info:
```

```
 Data:
Target Node Name: 50:0a:09:80:80:a2:ce:4d
Target Port Name: 50:0a:09:81:90:a2:ce:4d
Target Port ID: 01:12:00
Target Number: 0
Vendor ID: NETAPP
Product ID: LUN
Product Revision: 820a
Product Type: FCP Disk
Serial Number: 80ADI+KeW-ca
Lun Count: 1
Luns: LUN_1
Remote Type: Unknown
Session State: NA
Secure Mode: Disabled
Congestion Current State: Healthy
Virtual Lane: Normal
I/O Throttling: None
Status: Online
```

The following example shows the `target info get` command for NVMe targets on a local host:

```
~# esxcli qlfc qcc target info get -i
FC_51:40:2e:c0:12:3c:f4:a2_Target_20:02:00:11:0d:27:b2:00
```

The following example shows the `target info get` command for NVME targets on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc target info
get -i FC_51:40:2e:c0:12:3c:f4:a2_Target_20:02:00:11:0d:27:b2:00
```

The output for both commands is:

```
Target Information:
   Message: Success
   Target Info:
         Data:
      Target Node Name: 20:02:00:11:0d:27:b2:00
      Target Port Name: 20:02:00:11:0d:27:b2:00
      Target Port ID: 01:00:00
      Target Number: 1
      Vendor ID: SANBlaze
      Product ID: VLUN P2T0
```

```
Product Revision: V8.1-dev
Product Type: NVMe Disk
Serial Number: 00-200200110d27b200
Storage NQN: nqn.2014-08.com.sanblaze:virtualun.
             virtuallunnvme.2.0
Storage Controller ID: 0x103
Namespace Count: 1
Namespaces: NSID_1
Remote Type: Unknown
Session State: NA
Secure Mode: Disabled
Congestion Current State: Healthy
Virtual Lane: Normal
I/O Throttling: None
Status: Online
```

Adapter Target Universal SAN Congestion Mitigation Status

For a description of USCM, see the following NOTE.

The `target scmchk` command displays USCM status of the selected target discovered on the Marvell Adapter target:

`~# esxcli qlfc qcc target scmchk`

Syntax

```
esxcli qlfc qcc target scmchk {cmd} [cmd options]
```

Available Commands

`get`
Shows USCM status information of the specified target. See Get Target USCM Status.

Get Target USCM Status

The `target scmchk get` command displays target-level USCM status of the selected target discovered on the Marvell Adapter port:

`~# esxcli qlfc qcc target scmchk get`

Syntax

```
esxcli qlfc qcc target scmchk get [cmd options]
```

Keywords

`-i|--identifier <str>` = Target identifier (required). Obtain the identifier from the target `list` output.

Example

The following example shows the `target scmchk get` command on a local host:

```
~# esxcli qlfc qcc target scmchk get -i
FC_51:40:2e:c0:12:3c:f4:a0_Target_50:0a:09:81:90:a2:ce:4d
```

The following example shows the `target scmchk get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc target scmchk
get -i FC_51:40:2e:c0:12:3c:f4:a0_Target_50:0a:09:81:90:a2:ce:4d
```

The output for both commands is:

```
Target Congestion Information:
   Message: Success
   Target Congestion Status:
         Data:
      Node Name: 50:0a:09:80:80:a2:ce:4d
      Port Name: 50:0a:09:81:90:a2:ce:4d
      Port Id: 01:12:00
      Target Number: 0
      Congestion Current State: Healthy
      Link Integrity Events: No
      Seconds Since Last Event: 153
      Virtual Lane: Normal
      I/O Throttling: None
```

Target Port Status

The USCM target congestion status (Table 4-4) indicates the current status/heath of connected target ports based on the FPIN ELSs received from the switch.

This status includes peer congestion and link integrity, as well as details about each of these events for each active target port.

Table 4-4. USCM Target Port Status

Status	Description
`Congestion Current State`	See Congestion Current State
`Link Integrity Events`	**`Yes.`** A link event has been received. **`No`**
`Seconds Since Last Event`	See Seconds Since Last Event.
`Virtual Lane` [a]	`Slow`. The current target is marked as slow device and was moved to a slow virtual lane due to congestion. `Normal`. The current target is healthy. No congestion is detected. `Fast`. Fast traffic is flowing to a fast virtual lane. `Non-operational`. Either virtual lanes is disabled at the initiator or it is not supported by the connected switch. `NA`. Virtual lanes is not supported on the connected adapter and/or the driver.
`I/O Throttling` [a]	See I/O Throttling

[a] For more information about virtual lane and I/O throttling, see the appropriate Marvell QLogic Fibre Channel Adapter user's guide.

Adapter Target Universal SAN Congestion Mitigation Statistics

For a description of USCM, see the following NOTE.

For more information about USCM statistics, see "Get Initiator Port USCM Statistics" on page 222.

The `target scmstats` command displays USCM statistics for the specified target discovered on the Marvell Adapter port:

```
~# esxcli qlfc qcc target scmstats
```

Syntax

```
esxcli qlfc qcc target scmstats {cmd} [cmd options]
```

Available Commands

`get`
Shows USCM statistics information of the specified target. See Get Target SAN Congestion Management (SCM) Statistics.

Get Target USCM Statistics

The `target scmstats get` command displays target-level USCM statistics for the specified target discovered on the Marvell Adapter port.

```
~# esxcli qlfc qcc target scmstats get
```

Syntax

```
esxcli qlfc qcc target scmstats get [cmd options]
```

Keywords

`-i|--identifier <str>` = Target identifier (required). Obtain the identifier from the target `list` output.

Example

The following example shows the `target scmstats get` command on a local host:

```
~# esxcli qlfc qcc port scmstats get -i
FC_51:40:2e:c0:12:3c:f4:a2_Target_50:0a:09:81:90:a2:ce:4d
```

The following example shows the `target scmstats get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc port
scsmstats get
-i FC_51:40:2e:c0:12:3c:f4:a2_Target_50:0a:09:81:90:a2:ce:4d
```

The output for both commands is:

```
Target SCM Statistics:
   Message: Success
   Target SCMStats:
         Data:
      Node Name: 50:0a:09:80:80:a2:ce:4d
      Port Name: 50:0a:09:81:90:a2:ce:4d
      Port Id: 01:12:00
      Target Number: 0
```

```
---------------------------------
Congestion Mitigation
---------------------------------
Cleared Congestion Count: 3
Throttled Up Count: 1
Throttled Down Count: 4
Bottomed Out Count: 0
Returned Busy Count: 12
---------------------------------
Rx Fabric Performance Impact
Notifications
---------------------------------
Link Failure Count: 0
Link Unknown Event Count: 0
Loss Of Sync Count: 0
Loss Of Signal Count: 0
Link Device Specific Event Count: 0
Primitive Sequence Protocol Error Count: 0
Invalid Transmission Word Count: 0
Invalid CRC Count: 0
Link Uncorrectable FEC Count: 0
Congestion Clear Count: 3
Congestion Lost Credit Count: 2
Congestion Credit Stall Count: 5
Congestion Oversubscription Count: 2
Congestion Device Specific Count: 0
Pun Count: 0
```

Target Port Statistics

The USCM target statistics (Table 4-5) indicate how many times a congestion event has occurred since the counters were reset.

Table 4-5. USCM Target Port Congestion Mitigation Statistics

Statistic	Description
`Cleared Congestion Count`	Counter for the number of times the congestion event was cleared for this target.
`Throttled Up Count`	See Throttled Up Count.

Table 4-5. USCM Target Port Congestion Mitigation Statistics (Continued)

Statistic	Description
`Throttled Down Count`	See Throttled Down Count.
`Bottomed Out Count`	See Bottom Out Count.
`Returned Busy Count`	See Returned Busy Count.
`Link Failure Count`	Counter for number of link failure events.
`Link Unknown Event Count`	Counter for the number of unknown events.
`Loss of Sync Count`	Counter for the number of loss of sync events.
`Loss of Signal Count`	Counter for the number of loss of signal count events.
`Link Device Specific Event Count`	Counter for the number of device specific events.
`Primitive Seq Protocol Error Count`	Counter for the number of primitive sequence protocol errors.
`Invalid Transmission Word Count`	Counter for the number of transmission word errors.
`Invalid CRC Count`	Counter for the number of invalid CRC events.
`Link Uncorrectable FEC Count`	Counter for the number of uncorrectable FEC errors.
`Congestion Clear Count`	Counter for the number of times the peer congestion event was cleared for this target.
`Congestion Lost Credit Count`	Counter for the number of lost credit events.
`Congestion Credit Stall Count`	Counter for the number of credit stall events.

Table 4-5. USCM Target Port Congestion Mitigation Statistics (Continued)

Statistic	Description
`Congestion Over-subscription Count`	Counter for the number of oversubscription events.
`Congestion Device Specific Count`	Counter for the number of device-specific events.
`PUN Count`	Number of priority update notifications received.

Adapter Target iiDMA Configuration

The `target iidma` command configures the maximum target link speed at which data is sent for the selected FC target from the Marvell Adapter port:

`~# esxcli qlfc qcc target iidma`

Syntax

```
esxcli qlfc qcc target iidma {cmd} [cmd options]
```

Available Commands

`get`
Shows current maximum link speed configuration of the specified target detected on Marvell FC Port. See Get Target iiDMA Configuration.

`set`
Sets maximum link speed configuration of the specified target detected on Marvell FC Port. See Set Target iiDMA Configuration.

Get Target iiDMA Configuration

The `target iidma get` command shows target link speed configuration for the specified FC target.

`~#` **`esxcli qlfc qcc target iidma get`**

Syntax

```
esxcli qlfc qcc target iidma get [cmd options]
```

Keywords

`-i|--identifier <str>` = Target identifier (required). Obtain the identifier from the target `list` output.

Example

The following example shows the `target iidma get` command on a local host:

```
~# esxcli qlfc qcc target iidma get -i
FC_51:40:2e:c0:12:3c:f4:A2_Target_50:0a:09:81:90:a2:ce:4d
```

The following example shows the `target iidma get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc target iidma
get -i FC_51:40:2e:c0:12:3c:f4:a2_Target_50:0a:09:81:90:a2:ce:4d
```

The output for both commands is:

```
Target IIDMA Information:
   Message: Success
   Target IIDMAInfo:
         Data:
      Node Name: 50:0a:09:80:80:a2:ce:4d
      Port Name: 50:0a:09:81:90:a2:ce:4d
      Port Id: 01:12:00
      Target Number: 0
      Target Maximum Link Speed: 16_Gbps
```

Set Target iiDMA Configuration

The `target iidma set` command sets the target link speed for the specified FC target.

```
~# esxcli qlfc qcc target iidma set
```

NOTE

The maximum link speed is limited to the maximum supported port speed shown in the `port info` command.

Syntax

```
esxcli qlfc qcc target iidma set [cmd options]
```

Keywords

`-i|--identifier <str>` = Target identifier (required). Obtain the identifier from the target `list` output.

`-s|--link-speed <str>` = Link speed (required)

Parameter	Value (strings are case-sensitive)
`identifier`	Port identifier from the port `list` command
`link-speed`	1_Gbps, 2_Gbps, 4_Gbps, 8_Gbps, 16_Gbps, 32_Gbps, 64_Gbps

Example

The following example shows the `target iidma set` command on a local host:

```
~# esxcli qlfc qcc target iidma set -i
FC_51:40:2e:c0:12:3c:f4:a2_Target_50:0a:09:81:90:a2:ce:4d -s
32_Gbps
```

The following example shows the `target iidma set` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc target iidma
set -i FC_51:40:2e:c0:12:3c:f4:a2_Target_50:0a:09:81:90:a2:ce:4d
-s 32_Gbps
```

The output for both commands is:

```
Target IIDMA Information:
   Message: Success
   Target IIDMAInfo:
         Data:
      Node Name: 50:0a:09:80:80:a2:ce:4d
      Port Name: 50:0a:09:81:90:a2:ce:4d
      Port Id: 01:12:00
Target Number: 0
      Target Maximum Link Speed: 32_Gbps
```

Target Link Statistics

The `target stats` command shows target link statistics for the selected FC target from the Marvell Adapter port:

```
~# esxcli qlfc qcc target stats
```

Syntax

```
esxcli qlfc qcc target stats {cmd} [cmd options]
```

Available Commands

`get`
Shows link statistics of the specified FC target detected on the Marvell FC port. See Get Target Link Statistics.

Get Target Link Statistics

The `target stats get` command shows target link statistics for the specified FC target detected on the Marvell FC port.

```
~# esxcli qlfc qcc target stats get
```

Syntax

```
esxcli qlfc qcc target stats get [cmd options]
```

Keywords

`-i|--identifier <str>` = Target identifier (required). Obtain the identifier from the target `list` output.

Example

The following example shows the `target stats get` command on a local host:

```
~# esxcli qlfc qcc target stats get -i
FC_51:40:2e:c0:12:3c:f4:a2_Target_50:0a:09:81:90:a2:ce:4d
```

The following example shows the `target stats get` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc target stats
get -i FC_51:40:2e:c0:12:3c:f4:a2_Target_50:0a:09:81:90:a2:ce:4d
```

The output for both commands is:

```
Target Statistics:
   Message: Success
   Target Stats:
         Data:
      Node Name: 50:0a:09:80:80:a2:ce:4d
      Port Name: 50:0a:09:81:90:a2:ce:4d
      Port Id: 0a:00:00
      Target Number: 0
      Link Failure Count: 1
```

```
Loss Of Sync Count: 0
Loss Of Signal Count: 0
Primitive Sequence Protocol Error Count: 0
Invalid Transmission Word Count: 0
Invalid CRC Count: 0
```

Adapter LUN Namespace Commands and Sub-namespaces

This section provides information on the following:

- Adapter LUN List
- Adapter LUN Information

NOTE

These commands are supported for FCP targets and NVMe-FC targets.

To view the Marvell Adapter LUN namespace commands and sub-namespaces, issue the following command:

```
# esxcli qlfc qcc lun
```

Syntax

```
esxcli qlfc qcc lun {cmd} [cmd options]
```

Namespaces

Available namespaces include the following:

Namespace	Description
`info`	QCC CLI for LUN information

Available Commands

`list`
Lists identifiers for the LUNs discovered on the Marvell Adapter ports. See Adapter LUN List.

Adapter LUN List

The `lun list` command shows all the LUNs detected, as follows:

```
~# esxcli qlfc qcc lun list
```

The following example shows the `lun list` command on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc lun list
```

The command output is:

```
Luns/Namespaces
------------------------------------------------------------
FC_51:40:2e:c0:12:3c:f4:a2_Target_50:0a:09:81:90:a2:ce:4d_LUN_0
FC_51:40:2e:c0:12:3c:f4:a2_Target_50:0a:09:81:90:a2:ce:4d_LUN_1
FC_51:40:2e:c0:12:3c:f4:a2_Target_ 20:02:00:11:0d:27:b2:00_NSID_1
```

Adapter LUN Information

The `lun info` command shows LUN information of the selected FC LUN:

`~# esxcli qlfc qcc lun info`

Syntax

`esxcli qlfc qcc lun info {cmd} [cmd options]`

Available Commands

`get`
Shows LUN information of the specified LUN. See Get LUN Information.

Get LUN Information

The `lun info get` command shows information for the specified LUN.

`~# esxcli qlfc qcc lun info get`

Syntax

`esxcli qlfc qcc lun info get [cmd options]`

Keywords

`-i|--identifier <str>` = LUN identifier (required). Obtain the identifier from the LUN `list` output.

Example

The following example shows the `lun info get` command for an FCP target on a local host:

`~# esxcli qlfc qcc lun info get -i FC_51:40:2e:c0:12:3c:f4:a2_Target_50:0a:09:81:90:a2:ce:4d_LUN_1`

The following example shows the `lun info get` command an FCP target on a remote host:

`# esxcli -s 172.27.9.144 -u root -p password qlfc qcc lun info get -i FC_51:40:2e:c0:12:3c:f4:a2_Target_50:0a:09:81:90:a2:ce:4d_LUN_1`

The output for both commands is:

```
Lun Information:
```

```
   Lun Info:
       Data:
      Vendor ID: NETAPP
      Product ID: LUN
      Product Revision: 820a
      Lun Number: 1
      Lun Size In MB: 3074
      Lun Worldwide Unique ID:
4e:45:54:41:50:50:20:20:20:4c:55:4e:20:38:30:41:44:49:2b:4b:65:57:
2d:63:61:00:00:00:00:00:00:00

Message: Success
```

The following example shows the `lun info get` command for an NVMe target on a local host:

```
~# esxcli qlfc qcc lun info get -i
FC_51:40:2e:c0:12:3c:f4:a2_Target_20:02:00:11:0d:27:b2:00_NSID_1
```

The following example shows the `lun info get` command for an NVMe target on a remote host:

```
# esxcli -s 172.27.9.144 -u root -p password qlfc qcc lun info get
-i FC_51:40:2e:c0:12:3c:f4:a2_Target_20:02:00:11:0d:27:b2:00_NSID_1
```

The output for both commands is:

```
Lun Information:
   Lun Info:
         Data:
      Vendor ID: SANBlaze
      Product ID: VLUN P2T0
      Product Revision: V8.1-dev
      Namespace ID: 1
      Namespace Size In MB: 5000

   Message: Success
```

A Revision History

Document Revision History	
Revision A, April 15, 2016	
Revision B, June 21, 2017	
Revision C, November 17, 2017	
Revision D, June 27, 2018	
Revision E, January 18, 2019	
Revision F, September 27, 2019	
Revision G, October 31, 2019	
Revision H, November 20, 2019	
Revision J, May 8, 2020	
Revision K, July 17, 2020	
Revision L, August 25, 2020	
Revision M, February 5, 2021	
Revision N, August 20, 2021	
Revision P, December 15, 2021	
Revision R, March 29, 2022	
Revision T, September 30, 2022	
Revision W, March 24, 2023	
Changes	**Sections Affected**
Removed support for VMware ESXi 6.0 and later. Added support for VMware ESXi 8.0 and later.	All
In the Keywords section: Added the `-m|--mode` keyword. Added the `identifier` and `mode` parameters. In the Example section: Updated the first two command examples to reflect Driver mode. Updated the corresponding output. Added two command examples for the command on a local and remote host for Firmware. Added the corresponding command output.	“Get Port Statistics” on page 152

Added the `port stats reset` command.	"Reset Port Statistics" on page 154
In the example output, updated the SFP DMI data display with 64GFC SFP related additional fields.	"Get Port SFP DMI Information" on page 155
In Table 4-1, I/O Throttling row, Description cell, **`Active`** value, changed the second sentence to "The I/O queue depth is ramping up **or down**."	"Initiator Port Status" on page 219
In the Example section, in the command output, `Rx Fabric Performance Impact Notifications` section, added a line for the `Link Uncorrectable FEC Count`.	"Get Initiator Port USCM Statistics" on page 222
In Table 4-3, added profiles for USCM Configuration Status and USCM Profile Management.	"Set Port Universal SAN Congestion Mitigation Profile" on page 227
In the Namespaces section, in the Namespace table, added the `stats` namespace.	"Adapter Target Namespace Commands and Sub-namespaces" on page 230
In the Example section, in the command output, `Rx Fabric Performance Impact Notifications` section, added a line for the `Link Uncorrectable FEC Count`.	"Get Target USCM Statistics" on page 237
In Table 4-3, added a new statistic, `Link Uncorrectable FEC Count`.	"Target Port Statistics" on page 238
Added the `target stats get` command to view target link statistics of the selected FC target.	"Target Link Statistics" on page 242, "Get Target Link Statistics" on page 243

Marvell first revolutionized the digital storage industry by moving information at speeds never thought possible. Today, that same breakthrough innovation remains at the heart of the company's storage, networking and connectivity solutions. With leading intellectual property and deep system-level knowledge, Marvell semiconductor solutions continue to transform the enterprise, cloud, automotive, industrial, and consumer markets. For more information, visit www.marvell.com.

Doc. No. BC0151101-00 Rev. W Revised: March 24, 2023