Managing Network Adapters

This chapter includes the following sections:

Overview of the Cisco UCS C-Series Network Adapters

Note The procedures in this chapter are available only when a Cisco UCS C-Series network adapter is installed in the chassis.

A Cisco UCS C-Series network adapter can be installed to provide options for I/O consolidation and virtualization support. The following adapters are available:

- Cisco UCS P81E Virtual Interface Card
- Cisco UCS VIC1225 Virtual Interface Card

The interactive *UCS Hardware and Software Interoperability Utility* lets you view the supported components and configurations for a selected server model and software release. The utility is available at the following URL: http://www.cisco.com/web/techdoc/ucs/interoperability/matrix/matrix.html

Cisco UCS P81E Virtual Interface Card

The Cisco UCS P81E Virtual Interface Card is optimized for virtualized environments, for organizations that seek increased mobility in their physical environments, and for data centers that want reduced costs through NIC, HBA, cabling, and switch reduction and reduced management overhead. This Fibre Channel over Ethernet (FCoE) PCIe card offers the following benefits:

- Allows up to 16 virtual Fibre Channel and 16 virtual Ethernet adapters to be provisioned in virtualized or nonvirtualized environments using just-in-time provisioning, providing tremendous system flexibility and allowing consolidation of multiple physical adapters.
- Delivers uncompromising virtualization support, including hardware-based implementation of Cisco VN-Link technology and pass-through switching.
- Improves system security and manageability by providing visibility and portability of network polices and security all the way to the virtual machine.

The virtual interface card makes Cisco VN-Link connections to the parent fabric interconnects, which allows virtual links to connect virtual NICs in virtual machines to virtual interfaces in the interconnect. In a Cisco Unified Computing System environment, virtual links then can be managed, network profiles applied, and interfaces dynamically reprovisioned as virtual machines move between servers in the system.

Cisco UCS VIC1225 Virtual Interface Card

The Cisco UCS VIC1225 Virtual Interface Card is a high-performance, converged network adapter that provides acceleration for the various new operational modes introduced by server virtualization. It brings superior flexibility, performance, and bandwidth to the new generation of Cisco UCS C-Series Rack-Mount Servers.

The Cisco UCS VIC 1225 implements the Cisco Virtual Machine Fabric Extender (VM-FEX), which unifies virtual and physical networking into a single infrastructure. It provides virtual-machine visibility from the physical network and a consistent network operations model for physical and virtual servers. In virtualized environments, this highly configurable and self-virtualized adapter provides integrated, modular LAN interfaces on Cisco UCS C-Series Rack-Mount Servers. Additional features and capabilities include:

- Supports up to 256 PCIe virtual devices, either virtual network interface cards (vNICs) or virtual host bus adapters (vHBAs), with high I/O operations per second (IOPS), support for lossless Ethernet, and 20 Gbps to servers.
- PCIe Gen2 x16 helps assure optimal bandwidth to the host for network-intensive applications with a redundant path to the fabric interconnect.
- Half-height design reserves full-height slots in servers for Cisco certified third-party adapters.
- Centrally managed by Cisco UCS Manager with support for Microsoft Windows, Red Hat Enterprise Linux, SUSE Linux, VMware vSphere, and Citrix XenServer.

Viewing Network Adapter Properties

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **show adapter** [*index*] [**detail**]	Displays adapter properties. To display the properties of a single adapter, specify the PCI slot number as the *index* argument.

This example displays the properties of adapter 2:

```
Server# scope chassis
Server /chassis # show adapter
PCI Slot Product Name   Serial Number  Product ID     Vendor
-------- -------------- -------------- -------------- --------------------
1        UCS VIC 1225   FCH1613796C    UCSC-PCIE-C... Cisco Systems Inc

Server /chassis # show adapter 2 detail
PCI Slot 2:
    Product Name: UCS VIC 1225
    Serial Number: FCH1613796C
    Product ID: UCSC-PCIE-CSC-02
    Adapter Hardware Revision: 4
    Current FW Version: 2.1(0.291)
    NIV: Disabled
    FIP: Enabled
    Configuration Pending: no
    CIMC Management Enabled : no
    VID: V00
    Vendor: Cisco Systems Inc
    Description:
    Bootloader Version: 2.1(0.291)
    FW Image 1 Version: 2.1(0.291)
    FW Image 1 State: RUNNING ACTIVATED
    FW Image 2 Version: 1.6(0.547)
    FW Image 2 State: BACKUP INACTIVATED
    FW Update Status: Idle
    FW Update Error: No error
    FW Update Stage: No operation (0%)
    FW Update Overall Progress: 0%

Server /chassis #
```

Configuring Network Adapter Properties

Before You Begin

- You must log in with admin privileges to perform this task.
- A supported Virtual Interface Card (VIC) must be installed in the chassis and the server must be powered on.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **show adapter**	(Optional) Displays the available adapter devices.
Step 3	Server /chassis # **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** The server must be powered on before you can view or change adapter settings.
Step 4	Server /chassis/adapter # **set fip-mode** {**disable** | **enable**}	Enables or disables FCoE Initialization Protocol (FIP) on the adapter card. FIP is enabled by default. **Note** We recommend that you disable this option only when explicitly directed to do so by a technical support representative.
Step 5	Server /chassis/adapter # **set niv-mode** {**disable** | **enable**}	Enables or disables Network Interface Virtualization (NIV) on the adapter card. NIV is disabled by default. If NIV mode is enabled, vNICs: • Can be assigned to a specific channel • Can be associated with a port profile • Can fail over to another vNIC if there are communication problems
Step 6	Server /chassis/adapter # **configure-vmfex** *port-count*	If NIV mode is enabled, *port-count* specifies the number of VM FEX interfaces you want CIMC to create, from 0 to 112.
Step 7	Server /chassis/adapter # **commit**	Commits the transaction to the system configuration.

This example configures the properties of adapter 1:

```
Server# scope chassis
Server /chassis # scope adapter 1
Server /chassis/adapter # set fip-mode enable
Server /chassis/adapter *# commit
Server /chassis/adapter #
```

Managing vHBAs

Guidelines for Managing vHBAs

When managing vHBAs, consider the following guidelines and restrictions:

- The Cisco UCS P81E Virtual Interface Card and Cisco UCS VIC1225 Virtual Interface Card provide two vHBAs (fc0 and fc1). You can create up to 16 additional vHBAs on these adapter cards.

Note If Network Interface Virtualization (NIV) mode is enabled for the adapter, you must assign a channel number to a vHBA when you create it.

- When using the Cisco UCS P81E Virtual Interface Card or Cisco UCS VIC1225 Virtual Interface Card in an FCoE application, you must associate the vHBA with the FCoE VLAN. Follow the instructions in Modifying vHBA Properties, on page 6 to assign the VLAN.
- After making configuration changes, you must reboot the host for settings to take effect.

Viewing vHBA Properties

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** The server must be powered on before you can view or change adapter settings.
Step 3	Server /chassis/adapter # **show host-fc-if** [**fc0** | **fc1** | *name*] [**detail**]	Displays properties of a single vHBA, if specified, or all vHBAs.

This example displays all vHBAs on adapter card 1 and the detailed properties of fc0:

```
Server# scope chassis
Server /chassis # scope adapter 1
Server /chassis/adapter # show host-fc-if
Name     World Wide Port Name     FC SAN Boot Uplink Port
-------- ----------------------- ----------- -----------
fc0      20:00:00:22:BD:D6:5C:35  Disabled    0
fc1      20:00:00:22:BD:D6:5C:36  Disabled    1

Server /chassis/adapter # show host-fc-if fc0 detail
Name fc0:
    World Wide Node Name: 10:00:00:22:BD:D6:5C:35
    World Wide Port Name: 20:00:00:22:BD:D6:5C:35
    FC SAN Boot: Disabled
    Persistent LUN Binding: Disabled
    Uplink Port: 0
    MAC Address: 00:22:BD:D6:5C:35
    CoS: 3
    VLAN: NONE
    Rate Limiting: OFF
    PCIe Device Order: ANY
    EDTOV: 2000
    RATOV: 10000
    Maximum Data Field Size: 2112
    Channel Number: 3
    Port Profile:

Server /chassis/adapter #
```

Modifying vHBA Properties

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **show adapter**	(Optional) Displays the available adapter devices.
Step 3	Server /chassis # **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** The server must be powered on before you can view or change adapter settings.
Step 4	Server /chassis/adapter # **scope host-fc-if** {**fc0** | **fc1** | *name*}	Enters the host Fibre Channel interface command mode for the specified vHBA.
Step 5	Server /chassis/adapter/host-fc-if # **set wwnn** *wwnn*	Specifies a unique World Wide Node Name (WWNN) for the adapter in the form hh:hh:hh:hh:hh:hh:hh:hh. Unless specified by this command, the WWNN is generated automatically by the system.
Step 6	Server /chassis/adapter/host-fc-if # **set wwpn** *wwpn*	Specifies a unique World Wide Port Name (WWPN) for the adapter in the form hh:hh:hh:hh:hh:hh:hh:hh. Unless specified by this command, the WWPN is generated automatically by the system.
Step 7	Server /chassis/adapter/host-fc-if # **set boot** {**disable** | **enable**}	Enables or disables FC SAN boot. The default is disable.
Step 8	Server /chassis/adapter/host-fc-if # **set persistent-lun-binding** {**disable** | **enable**}	Enables or disables persistent LUN binding. The default is disable.
Step 9	Server /chassis/adapter/host-fc-if # **set mac-addr** *mac-addr*	Specifies a MAC address for the vHBA.
Step 10	Server /chassis/adapter/host-fc-if # **set vlan** {**none** | *vlan-id*}	Specifies the default VLAN for this vHBA. Valid VLAN numbers are 1 to 4094; the default is none.
Step 11	Server /chassis/adapter/host-fc-if # **set cos** *cos-value*	Specifies the class of service (CoS) value to be marked on received packets unless the vHBA is configured to trust host CoS. Valid CoS values are 0 to 6; the default is 0. Higher values indicate more important traffic. This setting is not functional in NIV mode.

	Command or Action	Purpose
Step 12	Server /chassis/adapter/host-fc-if # **set rate-limit** {**off** | *rate*}	Specifies a maximum data rate for the vHBA. The range is 1 to 10000 Mbps; the default is off. This setting is not functional in NIV mode.
Step 13	Server /chassis/adapter/host-fc-if # **set order** {**any** | *0-99*}	Specifies the relative order of this device for PCIe bus device number assignment; the default is any.
Step 14	Server /chassis/adapter/host-fc-if # **set error-detect-timeout** *msec*	Specifies the error detect timeout value (EDTOV), the number of milliseconds to wait before the system assumes that an error has occurred. The range is 1000 to 100000; the default is 2000 milliseconds.
Step 15	Server /chassis/adapter/host-fc-if # **set resource-allocation-timeout** *msec*	Specifies the resource allocation timeout value (RATOV), the number of milliseconds to wait before the system assumes that a resource cannot be properly allocated. The range is 5000 to 100000; the default is 10000 milliseconds.
Step 16	Server /chassis/adapter/host-fc-if # **set max-field-size** *size*	Specifies the maximum size of the Fibre Channel frame payload (in bytes) that the vHBA supports. The range is 1 to 2112; the default is 2112 bytes.
Step 17	Server /chassis/adapter/host-fc-if # **scope error-recovery**	Enters the Fibre Channel error recovery command mode.
Step 18	Server /chassis/adapter/host-fc-if/error-recovery # **set fcp-error-recovery** {**disable** | **enable**}	Enables or disables FCP Error Recovery. The default is disable.
Step 19	Server /chassis/adapter/host-fc-if/error-recovery # **set link-down-timeout** *msec*	Specifies the link down timeout value, the number of milliseconds the uplink port should be offline before it informs the system that the uplink port is down and fabric connectivity has been lost. The range is 0 to 240000; the default is 30000 milliseconds.
Step 20	Server /chassis/adapter/host-fc-if/error-recovery # **set port-down-io-retry-count** *count*	Specifies the port down I/O retries value, the number of times an I/O request to a port is returned because the port is busy before the system decides the port is unavailable. The range is 0 to 255; the default is 8 retries.
Step 21	Server /chassis/adapter/host-fc-if/error-recovery # **set port-down-timeout** *msec*	Specifies the port down timeout value, the number of milliseconds a remote Fibre Channel port should be offline before informing the SCSI upper layer that the port is unavailable. The range is 0 to 240000; the default is 10000 milliseconds.
Step 22	Server /chassis/adapter/host-fc-if/error-recovery # **exit**	Exits to the host Fibre Channel interface command mode.

	Command or Action	Purpose
Step 23	Server /chassis/adapter/host-fc-if # **scope interrupt**	Enters the interrupt command mode.
Step 24	Server /chassis/adapter/host-fc-if/interrupt # **set interrupt-mode** {**intx** | **msi** | **msix**}	Specifies the Fibre Channel interrupt mode. The modes are as follows: • **intx —**Line-based interrupt (INTx) • **msi —**Message-Signaled Interrupt (MSI) • **msix —**Message Signaled Interrupts with the optional extension (MSIx). This is the recommended and default option.
Step 25	Server /chassis/adapter/host-fc-if/interrupt # **exit**	Exits to the host Fibre Channel interface command mode.
Step 26	Server /chassis/adapter/host-fc-if # **scope port**	Enters the Fibre Channel port command mode.
Step 27	Server /chassis/adapter/host-fc-if/port # **set outstanding-io-count** *count*	Specifies the I/O throttle count, the number of I/O operations that can be pending in the vHBA at one time. The range is 1 to 1024; the default is 512 operations.
Step 28	Server /chassis/adapter/host-fc-if/port # **set max-target-luns** *count*	Specifies the maximum logical unit numbers (LUNs) per target, the maximum number of LUNs that the driver will discover. This is usually an operating system platform limitation. The range is 1 to 1024; the default is 256 LUNs.
Step 29	Server /chassis/adapter/host-fc-if/port # **exit**	Exits to the host Fibre Channel interface command mode.
Step 30	Server /chassis/adapter/host-fc-if # **scope port-f-logi**	Enters the Fibre Channel fabric login command mode.
Step 31	Server /chassis/adapter/host-fc-if/port-f-logi # **set flogi-retries** {**infinite** | *count*}	Specifies the fabric login (FLOGI) retries value, the number of times that the system tries to log in to the fabric after the first failure. Enter a number between 0 and 4294967295 or enter **infinite** ; the default is infinite retries.
Step 32	Server /chassis/adapter/host-fc-if/port-f-logi # **set flogi-timeout** *msec*	Specifies the fabric login (FLOGI) timeout value, the number of milliseconds that the system waits before it tries to log in again. The range is 1 to 255000; the default is 2000 milliseconds.
Step 33	Server /chassis/adapter/host-fc-if/port-f-logi # **exit**	Exits to the host Fibre Channel interface command mode.

	Command or Action	Purpose
Step 34	Server /chassis/adapter/host-fc-if # **scope port-p-logi**	Enters the Fibre Channel port login command mode.
Step 35	Server /chassis/adapter/host-fc-if/port-p-logi # **set plogi-retries** *count*	Specifies the port login (PLOGI) retries value, the number of times that the system tries to log in to the fabric after the first failure. The range is 0 and 255; the default is 8 retries.
Step 36	Server /chassis/adapter/host-fc-if/port-p-logi # **set plogi-timeout** *msec*	Specifies the port login (PLOGI) timeout value, the number of milliseconds that the system waits before it tries to log in again. The range is 1 to 255000; the default is 2000 milliseconds.
Step 37	Server /chassis/adapter/host-fc-if/port-p-logi # **exit**	Exits to the host Fibre Channel interface command mode.
Step 38	Server /chassis/adapter/host-fc-if # **scope scsi-io**	Enters the SCSI I/O command mode.
Step 39	Server /chassis/adapter/host-fc-if/scsi-io # **set cdb-wq-count** *count*	The number of command descriptor block (CDB) transmit queue resources to allocate. The range is 1 to 8; the default is 1.
Step 40	Server /chassis/adapter/host-fc-if/scsi-io # **set cdb-wq-ring-size** *size*	The number of descriptors in the command descriptor block (CDB) transmit queue. The range is 64 to 512; the default is 512.
Step 41	Server /chassis/adapter/host-fc-if/scsi-io # **exit**	Exits to the host Fibre Channel interface command mode.
Step 42	Server /chassis/adapter/host-fc-if # **scope trans-queue**	Enters the Fibre Channel transmit queue command mode.
Step 43	Server /chassis/adapter/host-fc-if/trans-queue # **set fc-wq-ring-size** *size*	The number of descriptors in the Fibre Channel transmit queue. The range is 64 to 128; the default is 64.
Step 44	Server /chassis/adapter/host-fc-if/trans-queue # **exit**	Exits to the host Fibre Channel interface command mode.
Step 45	Server /chassis/adapter/host-fc-if # **scope recv-queue**	Enters the Fibre Channel receive queue command mode.
Step 46	Server /chassis/adapter/host-fc-if/recv-queue # **set fc-rq-ring-size** *size*	The number of descriptors in the Fibre Channel receive queue. The range is 64 to 128; the default is 64.
Step 47	Server /chassis/adapter/host-fc-if/recv-queue # **exit**	Exits to the host Fibre Channel interface command mode.

	Command or Action	Purpose
Step 48	Server /chassis/adapter/host-fc-if # **commit**	Commits the transaction to the system configuration. **Note** The changes will take effect upon the next server reboot.

This example configures the properties of a vHBA:

```
Server# scope chassis
Server /chassis # show adapter
PCI Slot Product Name   Serial Number  Product ID     Vendor
-------- -------------- -------------- -------------- --------------------
1        UCS VIC P81E   QCI1417A0QK    N2XX-ACPCI01   Cisco Systems Inc

Server /chassis # scope adapter 1
Server /chassis/adapter # scope host-fc-if fc1
Server /chassis/adapter/host-fc-if # set boot enable
Server /chassis/adapter/host-fc-if *# scope scsi-io
Server /chassis/adapter/host-fc-if/scsi-io *# set cdb-wq-count 2
Server /chassis/adapter/host-fc-if/scsi-io *# exit
Server /chassis/adapter/host-fc-if *# commit
Server /chassis/adapter/host-fc-if #
```

What to Do Next

Reboot the server to apply the changes.

Creating a vHBA

The adapter provides two permanent vHBAs. If NIV mode is enabled, you can create up to 16 additional vHBAs.

Before You Begin

You must log in with user or admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** The server must be powered on before you can view or change adapter settings.
Step 3	Server /chassis/adapter # **create host-fc-if** *name*	Creates a vHBA and enters the host Fibre Channel interface command mode. The *name* argument can be up to 32 ASCII characters.

	Command or Action	Purpose
Step 4	Server /chassis/adapter/host-fc-if # **set channel-number** *number*	(Optional) If NIV mode is enabled for the adapter, you must assign a channel number to this vHBA. The range is 1 to 1000.
Step 5	Server /chassis/adapter/host-fc-if # **commit**	Commits the transaction to the system configuration. **Note** The changes will take effect upon the next server reboot.

This example creates a vHBA on adapter 1:

```
Server# scope chassis
Server /chassis # scope adapter 1
Server /chassis/adapter # create host-fc-if Vhba5
Server /chassis/adapter/host-fc-if *# commit
New host-fc-if settings will take effect upon the next server reset
Server /chassis/adapter/host-fc-if #
```

What to Do Next

- Reboot the server to create the vHBA.
- If configuration changes are required, configure the new vHBA as described in Modifying vHBA Properties, on page 6.

Deleting a vHBA

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** The server must be powered on before you can view or change adapter settings.
Step 3	Server /chassis/adapter # **delete host-fc-if** *name*	Deletes the specified vHBA. **Note** You cannot delete either of the two default vHBAs, fc0 or fc1.
Step 4	Server /chassis/adapter # **commit**	Commits the transaction to the system configuration. **Note** The changes will take effect upon the next server reboot.

This example deletes a vHBA on adapter 1:

```
Server# scope chassis
Server /chassis # scope adapter 1
Server /chassis/adapter # delete host-fc-if Vhba5
Server /chassis/adapter *# commit
Server /chassis/adapter #
```

vHBA Boot Table

In the vHBA boot table, you can specify up to four LUNs from which the server can boot.

Viewing the Boot Table

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** The server must be powered on before you can view or change adapter settings.
Step 3	Server /chassis/adapter # **scope host-fc-if** {**fc0** | **fc1** | *name*}	Enters the host Fibre Channel interface command mode for the specified vHBA.
Step 4	Server /chassis/adapter/host-fc-if # **show boot**	Displays the boot table of the Fibre Channel interface.

This example displays the boot table for a vHBA:

```
Server# scope chassis
Server /chassis # scope adapter 1
Server /chassis/adapter # scope host-fc-if fc1
Server /chassis/adapter/host-fc-if # show boot
Boot Table Entry  Boot Target WWPN           Boot LUN ID
----------------- -------------------------- ------------
0                 20:00:00:11:22:33:44:55    3
1                 20:00:00:11:22:33:44:56    5

Server /chassis/adapter/host-fc-if #
```

Creating a Boot Table Entry

You can create up to four boot table entries.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** The server must be powered on before you can view or change adapter settings.
Step 3	Server /chassis/adapter # **scope host-fc-if** {**fc0** | **fc1** | *name*}	Enters the host Fibre Channel interface command mode for the specified vHBA.
Step 4	Server /chassis/adapter/host-fc-if # **create-boot-entry** *wwpn lun-id*	Creates a boot table entry. • *wwpn* — The World Wide Port Name (WWPN) for the boot target in the form hh:hh:hh:hh:hh:hh:hh:hh. • *lun-id* —The LUN ID of the boot LUN. The range is 0 to 255.
Step 5	Server /chassis/adapter/host-fc-if # **commit**	Commits the transaction to the system configuration. **Note** The changes will take effect upon the next server reboot.

This example creates a boot table entry for vHBA fc1:

```
Server# scope chassis
Server /chassis # scope adapter 1
Server /chassis/adapter # scope host-fc-if fc1
Server /chassis/adapter/host-fc-if # create-boot-entry 20:00:00:11:22:33:44:55 3
Server /chassis/adapter/host-fc-if *# commit
New boot table entry will take effect upon the next server reset
Server /chassis/adapter/host-fc-if #
```

Deleting a Boot Table Entry

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** The server must be powered on before you can view or change adapter settings.

	Command or Action	Purpose
Step 3	Server /chassis/adapter # **scope host-fc-if** {**fc0** | **fc1** | *name*}	Enters the host Fibre Channel interface command mode for the specified vHBA.
Step 4	Server /chassis/adapter/host-fc-if # **show boot**	Displays the boot table. From the Boot Table Entry field, locate the number of the entry to be deleted.
Step 5	Server /chassis/adapter/host-fc-if # **delete boot** *entry*	Deletes the boot table entry at the specified position in the table. The range of *entry* is 0 to 3. The change will take effect upon the next server reset.
Step 6	Server /chassis/adapter/host-fc-if # **commit**	Commits the transaction to the system configuration. **Note** The changes will take effect upon the next server reboot.

This example deletes boot table entry number 1 for the vHBA fc1:

```
Server# scope chassis
Server /chassis # scope adapter 1
Server /chassis/adapter # scope host-fc-if fc1
Server /chassis/adapter/host-fc-if # show boot
Boot Table Entry  Boot Target WWPN           Boot LUN ID
----------------- ------------------------- ------------
0                 20:00:00:11:22:33:44:55    3
1                 20:00:00:11:22:33:44:56    5

Server /chassis/adapter/host-fc-if # delete boot 1
Server /chassis/adapter/host-fc-if *# commit
New host-fc-if settings will take effect upon the next server reset
Server /chassis/adapter/host-fc-if # show boot
Boot Table Entry  Boot Target WWPN           Boot LUN ID
----------------- ------------------------- ------------
0                 20:00:00:11:22:33:44:55    3

Server /chassis/adapter/host-fc-if #
```

What to Do Next

Reboot the server to apply the changes.

vHBA Persistent Binding

Persistent binding ensures that the system-assigned mapping of Fibre Channel targets is maintained after a reboot.

Enabling Persistent Binding

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** The server must be powered on before you can view or change adapter settings.
Step 3	Server /chassis/adapter # **scope host-fc-if** {**fc0** | **fc1** | *name*}	Enters the host Fibre Channel interface command mode for the specified vHBA.
Step 4	Server /chassis/adapter/host-fc-if # **scope perbi**	Enters the persistent binding command mode for the vHBA.
Step 5	Server /chassis/adapter/host-fc-if/perbi # **set persistent-lun-binding enable**	Enables persistent binding for the vHBA.
Step 6	Server /chassis/adapter/host-fc-if/perbi # **commit**	Commits the transaction to the system configuration.

This example enables persistent binding for a vHBA:

```
Server# scope chassis
Server /chassis # scope adapter 4
Server /chassis/adapter # scope host-fc-if fc1
Server /chassis/adapter/host-fc-if # scope perbi
Server /chassis/adapter/host-fc-if/perbi # set persistent-lun-binding enable
Server /chassis/adapter/host-fc-if/perbi *# commit
Server /chassis/adapter/host-fc-if/perbi #
```

Disabling Persistent Binding

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** The server must be powered on before you can view or change adapter settings.

	Command or Action	Purpose
Step 3	Server /chassis/adapter # **scope host-fc-if** {**fc0** | **fc1** | *name*}	Enters the host Fibre Channel interface command mode for the specified vHBA.
Step 4	Server /chassis/adapter/host-fc-if # **scope perbi**	Enters the persistent binding command mode for the vHBA.
Step 5	Server /chassis/adapter/host-fc-if/perbi # **set persistent-lun-binding disable**	Disables persistent binding for the vHBA.
Step 6	Server /chassis/adapter/host-fc-if/perbi # **commit**	Commits the transaction to the system configuration.

This example disables persistent binding for a vHBA:

```
Server# scope chassis
Server /chassis # scope adapter 4
Server /chassis/adapter # scope host-fc-if fc1
Server /chassis/adapter/host-fc-if # scope perbi
Server /chassis/adapter/host-fc-if/perbi # set persistent-lun-binding disable
Server /chassis/adapter/host-fc-if/perbi *# commit
Server /chassis/adapter/host-fc-if/perbi #
```

Rebuilding Persistent Binding

Before You Begin

Persistent binding must be enabled in the vHBA properties.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** The server must be powered on before you can view or change adapter settings.
Step 3	Server /chassis/adapter # **scope host-fc-if** {**fc0** | **fc1** | *name*}	Enters the host Fibre Channel interface command mode for the specified vHBA.
Step 4	Server /chassis/adapter/host-fc-if # **scope perbi**	Enters the persistent binding command mode for the vHBA.
Step 5	Server /chassis/adapter/host-fc-if/perbi # **rebuild**	Rebuilds the persistent binding table for the vHBA.

This example rebuilds the persistent binding table for a vHBA:

```
Server# scope chassis
Server /chassis # scope adapter 4
Server /chassis/adapter # scope host-fc-if fc1
Server /chassis/adapter/host-fc-if # scope perbi
Server /chassis/adapter/host-fc-if/perbi # rebuild

Server /chassis/adapter/host-fc-if/perbi #
```

Managing vNICs

Guidelines for Managing vNICs

When managing vNICs, consider the following guidelines and restrictions:

- The Cisco UCS P81E Virtual Interface Card and Cisco UCS VIC1225 Virtual Interface Card provide two default vNICs (eth0 and eth1). You can create up to 16 additional vNICs on these adapter cards.

  Note If Network Interface Virtualization (NIV) mode is enabled for the adapter, you must assign a channel number to a vNIC when you create it.

- After making configuration changes, you must reboot the host for settings to take effect.

Viewing vNIC Properties

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** The server must be powered on before you can view or change adapter settings.
Step 3	Server /chassis/adapter # **show host-eth-if** [**eth0** | **eth1** | *name*] [**detail**]	Displays properties of a single vNIC, if specified, or all vNICs.

This example displays the brief properties of all vNICs and the detailed properties of eth0:

```
Server# scope chassis
Server /chassis # scope adapter 1
Server /chassis/adapter # show host-eth-if
Name     MTU  Uplink Port MAC Address       CoS VLAN PXE Boot
-------- ---- ----------- ----------------- --- ---- --------
eth0     1500 0           00:22:BD:D6:5C:33 0   NONE Enabled
eth1     1500 1           00:22:BD:D6:5C:34 0   NONE Enabled
```

```
Server /chassis/adapter # show host-eth-if eth0 detail
Name eth0:
    MTU: 1500
    Uplink Port: 0
    MAC Address: 00:22:BD:D6:5C:33
    CoS: 0
    Trust Host CoS: disabled
    PCI Order: ANY
    VLAN: NONE
    VLAN Mode: TRUNK
    Rate Limiting: OFF
    PXE Boot: enabled
    Channel Number: N/A
    Port Profile: N/A
    Uplink Failover: N/A
    Uplink Failback Timeout: N/A

Server /chassis/adapter #
```

Modifying vNIC Properties

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **show adapter**	(Optional) Displays the available adapter devices.
Step 3	Server /chassis # **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** The server must be powered on before you can view or change adapter settings.
Step 4	Server /chassis/adapter # **scope host-eth-if** {**eth0** | **eth1** | *name*}	Enters the host Ethernet interface command mode for the specified vNIC.
Step 5	Server /chassis/adapter/host-eth-if # **set mtu** *mtu-value*	Specifies the maximum transmission unit (MTU) or packet size that the vNIC accepts. Valid MTU values are 1500 to 9000 bytes; the default is 1500.
Step 6	Server /chassis/adapter/host-eth-if # **set uplink** {**0** | **1**}	Specifies the uplink port associated with this vNIC. All traffic for this vNIC goes through this uplink port.
Step 7	Server /chassis/adapter/host-eth-if # **set mac-addr** *mac-addr*	Specifies a MAC address for the vNIC in the form hh:hh:hh:hh:hh:hh or hhhh:hhhh:hhhh.
Step 8	Server /chassis/adapter/host-eth-if # **set cos** *cos-value*	Specifies the class of service (CoS) value to be marked on received packets unless the vNIC is configured to trust host CoS. Valid CoS values are 0 to 6; the default is 0. Higher values indicate more important traffic.

	Command or Action	Purpose
		Note If NIV is enabled, this setting is determined by the switch, and the command is ignored.
Step 9	Server /chassis/adapter/host-eth-if# **set trust-host-cos** {**disable** | **enable**}	Specifies whether the vNIC will trust host CoS or will remark packets. The behavior is as follows: • **disable —**Received packets are remarked with the configured CoS. This is the default. • **enable —**The existing CoS value of received packets (host CoS) is preserved.
Step 10	Server /chassis/adapter/host-eth-if# **set order** {**any** | *0-99*}	Specifies the relative order of this device for PCI bus device number assignment; the default is any.
Step 11	Server /chassis/adapter/host-eth-if# **set vlan** {**none** | *vlan-id*}	Specifies the default VLAN for this vNIC. Valid VLAN numbers are 1 to 4094; the default is none. **Note** If NIV is enabled, this setting is determined by the switch, and the command is ignored.
Step 12	Server /chassis/adapter/host-eth-if# **set vlan-mode** {**access** | **trunk**}	Specifies the VLAN mode for the vNIC. The modes are as follows: • **access —**The vNIC belongs to only one VLAN. • **trunk —**The vNIC can belong to more than one VLAN. This is the default. **Note** If NIV is enabled, this setting is determined by the switch, and the command is ignored.
Step 13	Server /chassis/adapter/host-eth-if# **set rate-limit** {**off** | *rate*}	Specifies a maximum data rate for the vNIC. The range is 1 to 10000 Mbps; the default is off. **Note** If NIV is enabled, this setting is determined by the switch, and the command is ignored.
Step 14	Server /chassis/adapter/host-eth-if# **set boot** {**disable** | **enable**}	Specifies whether the vNIC can be used to perform a PXE boot. The default is enable for the two default vNICs, and disable for user-created vNICs.
Step 15	Server /chassis/adapter/host-eth-if# **set channel-number** *number*	If NIV mode is enabled for the adapter, select the channel number that will be assigned to this vNIC. The range is 1 to 1000.
Step 16	Server /chassis/adapter/host-eth-if# **set port-profile** *name*	If NIV mode is enabled for the adapter, select the port profile that should be associated with the vNIC. **Note** The *name* must be a port profile defined on the switch to which this server is connected.
Step 17	Server /chassis/adapter/host-eth-if# **set uplink-failover** {**disable** | **enable**}	If NIV mode is enabled for the adapter, enable this setting if traffic on this vNIC should fail over to the

	Command or Action	Purpose
		secondary interface if there are communication problems.
Step 18	Server /chassis/adapter/host-eth-if # **set uplink-failback-timeout** *seconds*	After a vNIC has started using its secondary interface, this setting controls how long the primary interface must be available before the system resumes using the primary interface for the vNIC. Enter a number of *seconds* between 0 and 600.
Step 19	Server /chassis/adapter/host-eth-if # **scope interrupt**	Enters the interrupt command mode.
Step 20	Server /chassis/adapter/host-eth-if/interrupt # **set interrupt-count** *count*	Specifies the number of interrupt resources. The range is 1 to 514; the default is 8. In general, you should allocate one interrupt resource for each completion queue.
Step 21	Server /chassis/adapter/host-eth-if/interrupt # **set coalescing-time** *usec*	The time to wait between interrupts or the idle period that must be encountered before an interrupt is sent. The range is 1 to 65535 microseconds; the default is 125. To turn off coalescing, enter 0 (zero).
Step 22	Server /chassis/adapter/host-eth-if/interrupt # **set coalescing-type** {**idle** | **min**}	The coalescing types are as follows: • **idle —**The system does not send an interrupt until there is a period of no activity lasting as least as long as the time specified in the coalescing time configuration. • **min —**The system waits for the time specified in the coalescing time configuration before sending another interrupt event. This is the default.
Step 23	Server /chassis/adapter/host-eth-if/interrupt # **set interrupt-mode** {**intx** | **msi** | **msix**}	Specifies the Ethernet interrupt mode. The modes are as follows: • **intx —**Line-based interrupt (PCI INTx) • **msi —**Message-Signaled Interrupt (MSI) • **msix —**Message Signaled Interrupts with the optional extension (MSI-X). This is the recommended and default option.
Step 24	Server /chassis/adapter/host-eth-if/interrupt # **exit**	Exits to the host Ethernet interface command mode.
Step 25	Server /chassis/adapter/host-eth-if # **scope recv-queue**	Enters receive queue command mode.

	Command or Action	Purpose
Step 26	Server /chassis/adapter/host-eth-if/recv-queue # **set rq-count** *count*	The number of receive queue resources to allocate. The range is 1 to 256; the default is 4.
Step 27	Server /chassis/adapter/host-eth-if/recv-queue # **set rq-ring-size** *size*	The number of descriptors in the receive queue. The range is 64 to 4094; the default is 512.
Step 28	Server /chassis/adapter/host-eth-if/recv-queue # **exit**	Exits to the host Ethernet interface command mode.
Step 29	Server /chassis/adapter/host-eth-if # **scope trans-queue**	Enters transmit queue command mode.
Step 30	Server /chassis/adapter/host-eth-if/trans-queue # **set wq-count** *count*	The number of transmit queue resources to allocate. The range is 1 to 256; the default is 1.
Step 31	Server /chassis/adapter/host-eth-if/trans-queue # **set wq-ring-size** *size*	The number of descriptors in the transmit queue. The range is 64 to 4094; the default is 256.
Step 32	Server /chassis/adapter/host-eth-if/trans-queue # **exit**	Exits to the host Ethernet interface command mode.
Step 33	Server /chassis/adapter/host-eth-if # **scope comp-queue**	Enters completion queue command mode.
Step 34	Server /chassis/adapter/host-eth-if/comp-queue # **set cq-count** *count*	The number of completion queue resources to allocate. The range is 1 to 512; the default is 5. In general, the number of completion queues equals the number of transmit queues plus the number of receive queues.
Step 35	Server /chassis/adapter/host-eth-if/comp-queue # **exit**	Exits to the host Ethernet interface command mode.
Step 36	Server /chassis/adapter/host-eth-if # **scope offload**	Enters TCP offload command mode.
Step 37	Server /chassis/adapter/host-eth-if/offload # **set tcp-segment-offload** {**disable** | **enable**}	Enables or disables TCP Segmentation Offload as follows: • **disable —**The CPU segments large TCP packets. • **enable —**The CPU sends large TCP packets to the hardware to be segmented. This option may reduce CPU overhead and increase throughput rate. This is the default. **Note** This option is also known as Large Send Offload (LSO).

	Command or Action	Purpose
Step 38	Server /chassis/adapter/host-eth-if/offload # **set tcp-rx-checksum-offload** {**disable** | **enable**}	Enables or disables TCP Receive Offload Checksum Validation as follows: • **disable —**The CPU validates all packet checksums. • **enable —**The CPU sends all packet checksums to the hardware for validation. This option may reduce CPU overhead. This is the default.
Step 39	Server /chassis/adapter/host-eth-if/offload # **set tcp-tx-checksum-offload** {**disable** | **enable**}	Enables or disables TCP Transmit Offload Checksum Validation as follows: • **disable —**The CPU validates all packet checksums. • **enable —**The CPU sends all packet checksums to the hardware for validation. This option may reduce CPU overhead. This is the default.
Step 40	Server /chassis/adapter/host-eth-if/offload # **set tcp-large-receive-offload** {**disable** | **enable**}	Enables or disables TCP Large Packet Receive Offload as follows: • **disable —**The CPU processes all large packets. • **enable —**The hardware reassembles all segmented packets before sending them to the CPU. This option may reduce CPU utilization and increase inbound throughput. This is the default.
Step 41	Server /chassis/adapter/host-eth-if/offload # **exit**	Exits to the host Ethernet interface command mode.
Step 42	Server /chassis/adapter/host-eth-if # **scope rss**	Enters Receive-side Scaling (RSS) command mode.
Step 43	Server /chassis/adapter/host-eth-if/rss # **set rss** {**disable** | **enable**}	Enables or disables RSS, which allows the efficient distribution of network receive processing across multiple CPUs in multiprocessor systems. The default is enable for the two default vNICs, and disable for user-created vNICs.
Step 44	Server /chassis/adapter/host-eth-if/rss # **set rss-hash-ipv4** {**disable** | **enable**}	Enables or disables IPv4 RSS. The default is enable.
Step 45	Server /chassis/adapter/host-eth-if/rss # **set rss-hash-tcp-ipv4** {**disable** | **enable**}	Enables or disables TCP/IPv4 RSS. The default is enable.
Step 46	Server /chassis/adapter/host-eth-if/rss # **set rss-hash-ipv6** {**disable** | **enable**}	Enables or disables IPv6 RSS. The default is enable.

	Command or Action	Purpose
Step 47	Server /chassis/adapter/host-eth-if/rss # **set rss-hash-tcp-ipv6** {**disable** | **enable**}	Enables or disables TCP/IPv6 RSS. The default is enable.
Step 48	Server /chassis/adapter/host-eth-if/rss # **set rss-hash-ipv6-ex** {**disable** | **enable**}	Enables or disables IPv6 Extension RSS. The default is disable.
Step 49	Server /chassis/adapter/host-eth-if/rss # **set rss-hash-tcp-ipv6-ex** {**disable** | **enable**}	Enables or disables TCP/IPv6 Extension RSS. The default is disable.
Step 50	Server /chassis/adapter/host-eth-if/rss # **exit**	Exits to the host Ethernet interface command mode.
Step 51	Server /chassis/adapter/host-eth-if # **commit**	Commits the transaction to the system configuration. **Note** The changes will take effect upon the next server reboot.

This example configures the properties of a vNIC:

```
Server# scope chassis
Server /chassis # show adapter
PCI Slot Product Name   Serial Number  Product ID     Vendor
-------- -------------- -------------- -------------- --------------------
1        UCS VIC P81E   QCI1417A0QK    N2XX-ACPCI01   Cisco Systems Inc

Server /chassis # scope adapter 1
Server /chassis/adapter # scope host-eth-if Test1
Server /chassis/adapter/host-eth-if # set uplink 1
Server /chassis/adapter/host-eth-if *# scope offload
Server /chassis/adapter/host-eth-if/offload *# set tcp-segment-offload enable
Server /chassis/adapter/host-eth-if/offload *# exit
Server /chassis/adapter/host-eth-if *# commit
Server /chassis/adapter/host-eth-if #
```

What to Do Next

Reboot the server to apply the changes.

Creating a vNIC

The adapter provides two permanent vNICs. You can create up to 16 additional vNICs.

Before You Begin

You must log in with user or admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** The server must be powered on before you can view or change adapter settings.
Step 3	Server /chassis/adapter # **create host-eth-if** *name*	Creates a vNIC and enters the host Ethernet interface command mode. The *name* argument can be up to 32 ASCII characters.
Step 4	Server /chassis/adapter/host-eth-if # **set channel-number** *number*	(Optional) If NIV mode is enabled for the adapter, you must assign a channel number to this vNIC. The range is 1 to 1000.
Step 5	Server /chassis/adapter/host-eth-if # **commit**	Commits the transaction to the system configuration. **Note** The changes will take effect upon the next server reboot.

This example creates a vNIC on adapter 1:

```
Server# scope chassis
Server /chassis # scope adapter 1
Server /chassis/adapter # create host-eth-if Vnic5
Server /chassis/adapter/host-eth-if *# commit
New host-eth-if settings will take effect upon the next server reset
Server /chassis/adapter/host-eth-if #
```

Deleting a vNIC

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** The server must be powered on before you can view or change adapter settings.
Step 3	Server /chassis/adapter # **delete host-eth-if** *name*	Deletes the specified vNIC. **Note** You cannot delete either of the two default vNICs, eth0 or eth1.
Step 4	Server /chassis/adapter # **commit**	Commits the transaction to the system configuration.

	Command or Action	Purpose
		Note The changes will take effect upon the next server reboot.

This example deletes a vNIC on adapter 1:

```
Server# scope chassis
Server /chassis # scope adapter 1
Server /chassis/adapter # delete host-eth-if Vnic5
Server /chassis/adapter *# commit
Server /chassis/adapter #
```

Creating Cisco usNIC Using the CIMC CLI

Note Even though several properties are listed for Cisco usNIC in the usNIC properties dialog box, you must configure only the following properties because the other properties are not currently being used.

- **cq-count**
- **rq-count**
- **tq-count**
- **usnic-count**

Before You Begin

You must log in to the CIMC CLI with administrator privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	server# **scope chassis**	Enters chassis command mode.
Step 2	server/chassis# **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** Make sure that the server is powered on before you attempt to view or change adapter settings. To view the index of the adapters configured on you server, use the **show adapter** command.
Step 3	server/chassis/adapter# **scope host-eth-if** {**eth0** | **eth1**}	Enters the command mode for the vNIC. Specify the Ethernet ID based on the number of vNICs that you have configured in your environment. For example, specify **eth0** if you configured only one vNIC.

	Command or Action	Purpose
Step 4	server/chassis/adapter/host-eth-if# **create usnic-config 0**	Creates a usNIC config and enters its command mode. Make sure that you always set the index value to 0. **Note** To create a Cisco usNIC for the first time for a given vNIC using the CIMC CLI, you must first create a **usnic-config**. Subsequently, you only need to scope into the **usnic-config** and modify the properties for Cisco usNIC. For more information about modifying Cisco usNIC properties, see Modifying a Cisco usNIC using the CIMC CLI, on page 28.
Step 5	server/chassis/adapter/host-eth-if/usnic-config# **set cq-count** *count*	Specifies the number of completion queue resources to allocate. We recommend that you set this value to 4. The number of completion queues equals the number of transmit queues plus the number of receive queues.
Step 6	server/chassis/adapter/host-eth-if/usnic-config# **set rq-count** *count*	Specifies the number of receive queue resources to allocate. MPI uses 2 receive queues per process. We recommend that you set this value to 2.
Step 7	server/chassis/adapter/host-eth-if/usnic-config# **set tq-count** *count*	Specifies the number of transmit queue resources to allocate. MPI uses 2 transmit queues per process. We recommend that you set this value to 2.
Step 8	server/chassis/adapter/host-eth-if/usnic-config# **set usnic-count** *number of usNICs* .	Specifies the number of Cisco usNICs to create. Each MPI process that is running on the server requires a dedicated Cisco usNIC. Therefore, you might need to create up to 64 Cisco usNICs to sustain 64 MPI processes running simultaneously. We recommend that you create at least as many Cisco usNICs, per Cisco usNIC-enabled vNIC, as the number of physical cores on your server. For example, if you have 8 physical cores on your server, create 8 Cisco usNICs.
Step 9	server/chassis/adapter/host-eth-if /usnic-config# **commit**	Commits the transaction to the system configuration. **Note** The changes take effect when the server is rebooted.
Step 10	server/chassis/adapter/host-eth-if/usnic-config# **exit**	Exits to host Ethernet interface command mode.

	Command or Action	Purpose
Step 11	server/chassis/adapter/host-eth-if# **exit**	Exits to adapter interface command mode.
Step 12	server/chassis/adapter# **exit**	Exits to chassis interface command mode.
Step 13	server/chassis# **exit**	Exits to server interface command mode.
Step 14	server# **scope bios**	Enters Bios command mode.
Step 15	server/bios# **scope advanced**	Enters the advanced settings of BIOS command mode.
Step 16	server/bios/advanced# **set IntelVTD Enabled**	Enables the Intel Virtualization Technology.
Step 17	server/bios/advanced# **set ATS Enabled**	Enables the Intel VT-d Address Translation Services (ATS) support for the processor.
Step 18	server/bios/advanced# **set CoherencySupport Enabled**	Enables Intel VT-d coherency support for the processor.
Step 19	server /bios/advanced# **commit**	Commits the transaction to the system configuration. **Note** The changes take effect when the server is rebooted.

This example shows how to configure Cisco usNIC properties:

```
Server # scope chassis
server /chassis # show adapter
server /chassis # scope adapter 2
server /chassis/adapter # scope host-eth-if eth0
server /chassis/adapter/host-eth-if # create usnic-config 0
server /chassis/adapter/host-eth-if/usnic-config *# set usnic-count 64
server /chassis/adapter/host-eth-if/usnic-config *# set cq-count 4
server /chassis/adapter/host-eth-if/usnic-config *# set rq-count 2
server /chassis/adapter/host-eth-if/usnic-config *# set tq-count 2
server /chassis/adapter/host-eth-if/usnic-config *# commit
Committed settings will take effect upon the next server reset
server /chassis/adapter/host-eth-if/usnic-config # exit
server /chassis/adapter/host-eth-if # exit
server /chassis/adapter # exit
server /chassis # exit
server # exit
server/bios # scope bios
server/bios/advanced # scope advanced
server/bios/advanced # set Intel VTD Enabled
server/bios/advanced # set ATS Enabled
server/bios/advanced # set CoherencySupport Enabled
server /chassis/adapter/host-eth-if/usnic-config # commit
Committed settings will take effect upon the next server reset
```

Modifying a Cisco usNIC using the CIMC CLI

Note Even though several properties are listed for Cisco usNIC in the usNIC properties dialog box, you must configure only the following properties because the other properties are not currently being used.

- **cq-count**
- **rq-count**
- **tq-count**
- **usnic-count**

Before You Begin

You must log in to the CIMC GUI with administrator privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	server# **scope chassis**	Enters chassis command mode.
Step 2	server/chassis# **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** Make sure that the server is powered on before you attempt to view or change adapter settings. To view the index of the adapters configured on you server, use the **show adapter** command.
Step 3	server/chassis/adapter# **scope host-eth-if** {**eth0** | **eth1**}	Enters the command mode for the vNIC. Specify the Ethernet ID based on the number of vNICs that you have configured in your environment. For example, specify **eth0** if you configured only one vNIC.
Step 4	server/chassis/adapter/host-eth-if# **scope usnic-config 0**	Enters the command mode for the usNIC. Make sure that you always set the index value as 0 to configure a Cisco usNIC.
Step 5	server/chassis/adapter/host-eth-if/usnic-config# **set cq-count** *count*	The number of completion queue resources to allocate. We recommend that you set this value to 4. The number of completion queues equals the number of transmit queues plus the number of receive queues.
Step 6	server/chassis/adapter/host-eth-if/usnic-config# **set rq-count** *count*	Specifies the number of receive queue resources to allocate. MPI uses two receive queues per

	Command or Action	Purpose
		process. We recommend that you set this value to 2.
Step 7	server/chassis/adapter/host-eth-if/usnic-config# **set tq-count** *count*	Specifies the number of transmit queue resources to allocate. MPI uses two transmit queues per process. We recommend that you set this value to 2.
Step 8	server/chassis/adapter/host-eth-if/usnic-config# **set usnic-count** *number of usNICs* .	Specifies the number of Cisco usNICs to create. Each MPI process running on the server requires a dedicated Cisco usNIC. Therefore, you might need to create up to 64 Cisco usNIC to sustain 64 MPI processes running simultaneously. We recommend that you create at least as many Cisco usNIC, per Cisco usNIC-enabled vNIC, as the number of physical cores on your server. For example, if you have 8 physical cores on your server, create 8 usNICs.
Step 9	server /chassis/adapter/host-eth-if /usnic-config# **commit**	Commits the transaction to the system configuration. **Note** The changes take effect when the server is rebooted.
Step 10	server/chassis/adapter/host-eth-if/usnic-config# **exit**	Exits to host Ethernet interface command mode.
Step 11	server/chassis/adapter/host-eth-if# **exit**	Exits to adapter interface command mode.
Step 12	server/chassis/adapter# **exit**	Exits to chassis interface command mode.
Step 13	server/chassis# **exit**	Exits to server interface command mode.
Step 14	server# **scope bios**	Enters BIOS command mode.
Step 15	server/bios# **scope advanced**	Enters the advanced settings of BIOS command mode.
Step 16	server/bios/advanced# **set IntelVTD Enabled**	Enables the Intel Virtualization Technology.
Step 17	server/bios/advanced# **set ATS Enabled**	Enables the Intel VT-d Address Translation Services (ATS) support for the processor.
Step 18	server/bios/advanced# **set CoherencySupport Enabled**	Enables Intel VT-d coherency support for the processor.
Step 19	server/bios/advanced# **commit**	Commits the transaction to the system configuration. **Note** The changes take effect when the server is rebooted.

This example shows how to configure Cisco usNIC properties:

```
server # scope chassis
server /chassis # show adapter
server /chassis # scope adapter 2
server /chassis/adapter # scope host-eth-if eth0
server /chassis/adapter/host-eth-if # scope usnic-config 0
server /chassis/adapter/host-eth-if/usnic-config # set usnic-count 64
server /chassis/adapter/host-eth-if/usnic-config # set cq-count 4
server /chassis/adapter/host-eth-if/usnic-config # set rq-count 2
server /chassis/adapter/host-eth-if/usnic-config # set tq-count 2
server /chassis/adapter/host-eth-if/usnic-config # commit
Committed settings will take effect upon the next server reset
server /chassis/adapter/host-eth-if/usnic-config # exit
server /chassis/adapter/host-eth-if # exit
server /chassis/adapter # exit
server /chassis # exit
server # exit
server/bios # scope bios
server/bios/advanced # scope advanced
server/bios/advanced # set Intel VTD Enabled
server/bios/advanced # set ATS Enabled
server/bios/advanced # set CoherencySupport Enabled
server /chassis/adapter/host-eth-if/usnic-config # commit
Committed settings will take effect upon the next server reset
```

Viewing usNIC Properties

Before You Begin

You must log in with admin privileges to perform this task.

usNIC must be configured on a vNIC.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** The server must be powered on before you can view or change adapter settings.
Step 3	Server /chassis/adapter # **scope host-eth-if** {**eth0** | **eth1** | *name*}	Enters the host Ethernet interface command mode for the specified vNIC.
Step 4	Server /chassis/adapter/host-eth-if # **show usnic-config** *index*	Displays the usNIC properties for a vNIC.

This example displays the usNIC properties for a vNIC:

```
Server # scope chassis
Server /chassis # scope adapter 1
Server /chassis/adapter # scope host-eth-if eth0
Server /chassis/adapter/host-eth-if # show usnic-config 0
Idx usNIC Count TQ Count RQ Count CQ Count TQ Ring Size RQ Ring Size Interrupt Count
```

```
--- ------------ -------- -------- -------- ------------ ------------ ---------------
0   113          2        2        4        256          512          4
Server /chassis/adapter/host-eth-if #
```

Deleting Cisco usNIC from a vNIC

Before You Begin

You must log in to CIMC CLI with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	server# **scope chassis**	Enters chassis command mode.
Step 2	server/chassis# **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** Make sure that the server is powered on before you attempt to view or change adapter settings. To view the index of the adapters configured on you server, use the **show adapter** command.
Step 3	server/chassis/adapter# **scope host-eth-if** {**eth0** | **eth1**}	Enters the command mode for the vNIC. Specify the Ethernet ID based on the number of vNICs that you have configured in your environment. For example, specify **eth0** if you configured only one vNIC.
Step 4	Server/chassis/adapter/host-eth-if# **delete usnic-config 0**	Deletes the Cisco usNIC configuration for the vNIC.
Step 5	Server/chassis/adapter/host-eth-if# **commit**	Commits the transaction to the system configuration **Note** The changes take effect when the server is rebooted.

This example shows how to delete the Cisco usNIC configuration for a vNIC:

```
server # scope chassis
server/chassis # show adapter
server/chassis # scope adapter 1
server/chassis/adapter # scope host-eth-if eth0
Server/chassis/adapter/host-eth-if # delete usnic-config 0
server/adapter/host-eth-if/iscsi-boot *# commit
New host-eth-if settings will take effect upon the next adapter reboot

server /adapter/host-eth-if/usnic-config #
```

Configuring iSCSI Boot Capability

Configuring iSCSI Boot Capability for vNICs

When the rack-servers are configured in a standalone mode, and when the VIC adapters are directly attached to the Nexus 5000 family of switches, you can configure these VIC adapters to boot the servers remotely from iSCSI storage targets. You can configure Ethernet vNICs to enable a rack server to load the host OS image from remote iSCSI target devices.

To configure the iSCSI boot capability on a vNIC:

- You must log in with admin privileges to perform this task.
- To configure a vNIC to boot a server remotely from an iSCSI storage target, you must enable the PXE boot option on the vNIC.

Note You can configure a maximum of 2 iSCSI vNICs for each host.

Configuring iSCSI Boot Capability on a vNIC

You can configure a maximum of 2 iSCSI vNICs for each host.

Before You Begin

- To configure a vNIC to boot a server remotely from an iSCSI storage target, you must enable the PXE boot option on the vNIC.
- You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** The server must be powered on before you can view or change adapter settings.
Step 3	Server /chassis/adapter # **scope host-eth-if** {**eth0** | **eth1** | *name*}	Enters the host Ethernet interface command mode for the specified vNIC.
Step 4	Server /chassis/adapter/host-eth-if # **create iscsi-boot** *index*	Creates the iSCSI boot index for the vNIC. At this moment, only 0 is allowed as the index.

	Command or Action	Purpose
Step 5	Server /chassis/adapter/host-eth-if/iscsi-boot* # **create iscsi-target** *index*	Creates an iSCSI target for the vNIC. The value can either be 0 or 1.
Step 6	Server /chassis/adapter/host-eth-if/iscsi-boot* # **set dhcp-net-settings enabled**	Enables the DHCP network settings for the iSCSI boot.
Step 7	Server /chassis/adapter/host-eth-if/iscsi-boot* # **set initiator-name** *string*	Sets the initiator name. It cannot be more than 223 characters.
Step 8	Server /chassis/adapter/host-eth-if/iscsi-boot* # **set dhcp-iscsi-settings enabled**	Enables the DHCP iSCSI settings.
Step 9	Server /chassis/adapter/host-eth-if/iscsi-boot* # **commit**	Commits the transaction to the system configuration. **Note** The changes will take effect upon the next server reboot.

This example shows how to configure the iSCSI boot capability for a vNIC:

```
Server # scope chassis
Server /chassis # scope adapter 1
Server /chassis/adapter # scope host-eth-if eth0
Server /chassis/adapter/host-eth-if # create iscsi-boot 0
Server /adapter/host-eth-if/iscsi-boot *# set dhcp-net-settings enabled
Server /adapter/host-eth-if/iscsi-boot *# set initiator-name iqn.2012-01.com.adser:abcde
Server /adapter/host-eth-if/iscsi-boot *# set dhcp-iscsi-settings enabled
Server /adapter/host-eth-if/iscsi-boot *# commit

New host-eth-if settings will take effect upon the next server reset
Server /adapter/host-eth-if/iscsi-boot #
```

Deleting an iSCSI Boot Configuration for a vNIC

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** The server must be powered on before you can view or change adapter settings.

	Command or Action	Purpose
Step 3	Server /chassis/adapter # **scope host-eth-if** {**eth0** | **eth1** | *name*}	Enters the host Ethernet interface command mode for the specified vNIC.
Step 4	Server /chassis/adapter/host-eth-if # **delete iscsi-boot 0**	Deletes the iSCSI boot capability for the vNIC.
Step 5	Server /chassis/adapter/host-eth-if* # **commit**	Commits the transaction to the system configuration **Note** The changes will take effect upon the next server reboot.

This example shows how to delete the iSCSI boot capability for a vNIC:

```
Server # scope chassis
Server /chassis # scope adapter 1
Server /chassis/adapter # scope host-eth-if eth0
Server /chassis/adapter/host-eth-if # delete iscsi-boot 0
Server /adapter/host-eth-if/iscsi-boot *# commit
New host-eth-if settings will take effect upon the next server reset

Server /adapter/host-eth-if/iscsi-boot #
```

Managing VM FEX

Virtual Machine Fabric Extender

Cisco Virtual Machine Fabric Extender (VM FEX) extends the (prestandard) IEEE 802.1Qbh port extender architecture to virtual machines. In this architecture, each VM interface is provided with a virtual Peripheral Component Interconnect Express (PCIe) device and a virtual port on a switch.

Viewing VM FEX Properties

Before You Begin

- The server must be powered on, or the properties will not display.
- A supported Virtual Interface Card (VIC) must be installed in the chassis and the server must be powered on.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*.

	Command or Action	Purpose
		Note The server must be powered on before you can view or change adapter settings.
Step 3	Server /chassis/adapter # **show vmfex** [**detail**]	Displays the general VM FEX properties. For field descriptions, see General Properties Settings, on page 36.
Step 4	Server /chassis/adapter # **scope vmfex** *name*	Enters the command mode for the specified VM FEX interface.
Step 5	Server /chassis/adapter/vmfex # **show interrupt** [**detail**]	Displays Ethernet interrupt settings. For field descriptions, see Ethernet Interrupt Settings, on page 37.
Step 6	Server /chassis/adapter/vmfex # **show recv-queue** [**detail**]	Displays Ethernet receive queue settings. For field descriptions, see Ethernet Receive Queue Settings, on page 38.
Step 7	Server /chassis/adapter/vmfex # **show trans-queue** [**detail**]	Displays Ethernet transmit queue settings. For field descriptions, see Ethernet Transmit Queue Settings, on page 38.
Step 8	Server /chassis/adapter/vmfex # **show comp-queue** [**detail**]	Displays completion queue settings. For field descriptions, see Completion Queue Settings, on page 38.
Step 9	Server /chassis/adapter/vmfex # **show offload** [**detail**]	Displays TCP offload settings. For field descriptions, see TCP Offload Settings, on page 38.
Step 10	Server /chassis/adapter/vmfex # **show rss** [**detail**]	Displays RSS settings. For field descriptions, see Receive Side Scaling Settings, on page 39.

This example displays the VM FEX properties:

```
Server /chassis/adapter # show vmfex detail
Name pts0:
    MTU: 1500
    Uplink Port: 0
    MAC Address: 00:00:00:00:00:00
    CoS: N/A
    Trust Host CoS:
    PCI Order:
    VLAN: N/A
    VLAN Mode: N/A
    Rate Limiting:
    PXE Boot: disabled
    Channel Number: 0
    Port Profile:
    Uplink Failover: Enabled
    Uplink Failback Timeout: 5

Server /chassis/adapter # scope vmfex pts0

Server /chassis/adapter/vmfex # show interrupt
Interrupt Count Coalescing Time (us) Coalescing Type Interrupt Mode
--------------- -------------------- --------------- ---------------
```

```
6               125                  MIN             MSI

Server /chassis/adapter/vmfex # show recv-queue
Receive Queue Count Receive Queue Ring Size
------------------- ------------------------------
4                   512

Server /chassis/adapter/vmfex # show trans-queue
Transmit Queue Count Transmit Queue Ring Size
-------------------- ------------------------------
1                    256

Server /chassis/adapter/vmfex # show comp-queue
Completion Queue Count    Completion Queue Ring Size
------------------------- --------------------------
5                         1

Server /chassis/adapter/vmfex # show offload
TCP Segment Offload  TCP Rx Checksum  TCP Tx Checksum  Large Receive
-------------------- ---------------- ---------------- -------------
enabled              enabled          enabled          enabled

Server /chassis/adapter/vmfex # show rss
TCP Rx Side Scaling
--------------------
enabled

Server /chassis/adapter/vmfex #
```

VM FEX Settings

The following tables describe the VM FEX settings that you can view.

General Properties Settings

Name	Description
Name	A user-defined name for the VM FEX.
MTU	The maximum transmission unit, or packet size, that this VM FEX accepts.
Uplink Port	The uplink port associated with this VM FEX. All traffic for this VM FEX goes through this uplink port.
MAC Address	The MAC address associated with the VM FEX.
Class of Service	The class of service to associate with traffic from this VM FEX.
Trust Host CoS	Whether the VM FEX can use the class of service provided by the host operating system.
PCI Order	The order in which this VM FEX will be used.
Default VLAN	The default VLAN for this VM FEX.
VLAN Mode	Whether VLAN trunking or access is configured.

Name	Description
Rate Limit	If rate limiting is configured, the maximum rate.
Enable PXE Boot	Whether the VM FEX can be used to perform a PXE boot.
Channel Number	If NIV mode is enabled for the adapter, the channel number assigned to this VM FEX.
Port Profile	If NIV mode is enabled for the adapter, the port profile associated with theVM FEX. **Note** This field displays the port profiles defined on the switch to which this server is connected.
Enable Uplink Failover	If NIV mode is enabled for the adapter, whether traffic on this VM FEX should fail over to the secondary interface if there are communication problems.
Failback Timeout	After a VM FEX has started using its secondary interface, this setting controls how long the primary interface must be available before the system resumes using the primary interface for the VM FEX.

Ethernet Interrupt Settings

Name	Description
Interrupt Count field	The number of interrupt resources allocated to this VM FEX.
Coalescing Time field	The time CIMC waits between interrupts or the idle period that must be encountered before an interrupt is sent.
Coalescing Type field	This can be one of the following: • **MIN—**The system waits for the time specified in the **Coalescing Time** field before sending another interrupt event. • **IDLE—**The system does not send an interrupt until there is a period of no activity lasting as least as long as the time specified in the **Coalescing Time** field.
Interrupt Mode field	The preferred driver interrupt mode. This can be one of the following: • **MSIx—**Message Signaled Interrupts (MSI) with the optional extension. • **MSI—**MSI only. • **INTx—**PCI INTx interrupts.

Ethernet Receive Queue Settings

Name	Description
Receive Queue Count field	The number of receive queue resources allocated to this VM FEX.
Receive Queue Ring Size field	The number of descriptors in each receive queue.

Ethernet Transmit Queue Settings

Name	Description
Transmit Queue Count field	The number of transmit queue resources allocated to this VM FEX.
Transmit Queue Ring Size field	The number of descriptors in each transmit queue.

Completion Queue Settings

Name	Description
Completion Queue Count field	The number of completion queue resources allocated to this VM FEX.
Completion Queue Ring Size field	The number of descriptors in each completion queue.

TCP Offload Settings

Name	Description
Enable TCP Segmentation Offload field	If enabled, the CPU sends large TCP packets to the hardware to be segmented. If disabled, the CPU segments large packets. **Note** This option is also known as Large Send Offload (LSO).
Enable TCP Rx Offload Checksum Validation field	If enabled, the CPU sends all packet checksums to the hardware for validation. If disabled, the CPU validates all packet checksums.
Enable TCP Tx Offload Checksum Generation field	If enabled, the CPU sends all packets to the hardware so that the checksum can be calculated. If disabled, the CPU calculates all packet checksums.
Enable Large Receive field	If enabled, the hardware reassembles all segmented packets before sending them to the CPU. If disabled, the CPU processes all large packets.

Receive Side Scaling Settings

Name	Description
Enable TCP Receive Side Scaling field	Receive Side Scaling (RSS) distributes network receive processing across multiple CPUs in multiprocessor systems. If enabled, network receive processing is shared across processors whenever possible. If disabled, network receive processing is always handled by a single processor even if additional processors are available.
Enable IPv4 RSS field	If enabled, RSS is enabled on IPv4 networks.
Enable TCP-IPv4 RSS field	If enabled, RSS is enabled for TCP transmissions across IPv4 networks.
Enable IPv6 RSS field	If enabled, RSS is enabled on IPv6 networks.
Enable TCP-IPv6 RSS field	If enabled, RSS is enabled for TCP transmissions across IPv6 networks.
Enable IPv6 Extension RSS field	If enabled, RSS is enabled for IPv6 extensions.
Enable TCP-IPv6 Extension RSS field	If enabled, RSS is enabled for TCP transmissions across IPv6 networks.

Managing Storage Adapters

Create Virtual Drive from Unused Physical Drives

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope storageadapter** *slot*	Enters command mode for an installed storage card.
Step 3	Server /chassis/storageadapter # **create virtual-drive**	At this point, you are prompted to enter information corresponding to the RAID level, the physical drives to be used, the size and the write policy for the new virtual drive. Enter the appropriate information at each prompt. When you have finished specifying the virtual drive information, you are prompted to confirm that the information is correct. Enter **y** (yes) to confirm, or **n** (no) to cancel the operation.

	Command or Action	Purpose
Step 4	Server /chassis/storageadapter # **show virtual-drive**	Displays the existing virtual drives.

This example shows how to create a new virtual drive that spans two unused physical drives.

```
Server# scope chassis
Server /chassis # scope storageadapter SLOT-3
Server /chassis/storageadapter # create-virtual-drive
Please enter RAID level
(0, 1, 5, 6, 10, 50, 60) --> 1

Please choose from the following 2 unused physical drives:
    Slot 6: size 68664 MB
    Slot 7: size 68664 MB

Specify physical disks for span 0:
  Enter comma-separated PDs from above list--> 6,7
Please enter Virtual Drive name (15 characters maximum)--> test_v_drive
Please enter Virtual Drive size in MB, GB, or TB
Example format: '400 GB' --> 1000 MB

Optional attribute:
  Write Policy: defaults to Write Back
    OK? (y or n)--> n
    0: Write Through
    1: Write Back
    2: Write Back with Bad BBU
    3: Write Back Adaptive
  Choose number from above options--> 2

Write Policy will be set to Write Back with Bad BBU (2 and 'write-policy\:2')

New virtual drive will have the following characteristics:
  - Spans: '[6.7]'
  - RAID level: '1'
  - Name: 'test_v_drive'
  - Size: 1000 MB
  - Write Policy: Write Back with Bad BBU
OK? (y or n)--> y

Server /chassis/storageadapter # show virtual-drive
Virtual Drive Health         Status               Name             Size       RAID Level
Boot Drive
------------- -------------- -------------------- ---------------- ---------- ----------
----------
0             Good           Optimal                               150528 MB  RAID 0
false
1             Good           Optimal                               20480 MB   RAID 0
true
2             Good           Optimal                               114140 MB  RAID 0
false
3             Good           Optimal              test_v_drive     1000 MB    RAID 1
false
4             Good           Optimal              new_from_test    500 MB     RAID 1
false

Server /chassis/storageadapter #
```

Create Virtual Drive from an Existing Drive Group

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope storageadapter** *slot*	Enters command mode for an installed storage card.
Step 3	Server /chassis/storageadapter # **carve-virtual-drive**	At this point, you are prompted to enter information corresponding to the virtual drives to be used, and the size and the write policy for the new virtual drive. Enter the appropriate information at each prompt. When you have finished specifying the virtual drive information, you are prompted to confirm that the information is correct. Enter **y** (yes) to confirm, or **n** (no) to cancel the operation.
Step 4	Server /chassis/storageadapter # **show virtual-drive**	Displays the existing virtual drives.

This example shows how to carve a new virtual drive out of unused space in an existing RAID 1 drive group:

```
Server# scope chassis
Server /chassis # scope storageadapter SLOT-3
Server /chassis/storageadapter # carve-virtual-drive
  < Fetching virtual drives...>

ID  Name              RL  VDSize        MaxPossibleSize PD(s)
-------------------------------------------------------------------
 3  test_v_drive      1   1000 MB       67664 MB        6,7

Please choose from the above list the virtual drive number
whose space the new virtual drive will share--> 3
New virtual drive will share space with VD 3

Please enter Virtual Drive name (15 characters maximum)--> new_from_test
Please enter Virtual Drive size in MB, GB, or TB (maximum: 67664 MB)
  Example format: '400 GB' --> 500 MB

Optional attribute:
  Write Policy: defaults to Write Back
    OK? (y or n)--> y

New virtual drive will have the following characteristics:
  - It will share space with virtual drive 3
  - Name: 'new_from_test'
  - Size: 500 MB

OK? (y or n)--> y
Server /chassis/storageadapter # show virtual-drive
Virtual Drive Health         Status               Name             Size       RAID Level
Boot Drive
```

```
------------ -------------- -------------------- ---------------- ---------- ----------
----------
0            Good           Optimal                                 150528 MB  RAID 0
false
1            Good           Optimal                                 20480 MB   RAID 0
true
2            Good           Optimal                                 114140 MB  RAID 0
false
3            Good           Optimal              test_v_drive       1000 MB    RAID 1
false
4            Good           Optimal              new_from_test      500 MB     RAID 1
false

Server /chassis/storageadapter #
```

Clearing Foreign Configuration

Important This task clears all foreign configuration on the controller. Also, all configuration information from all physical drives hosting foreign configuration is deleted. This action cannot be reverted.

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope storageadapter** *slot*	Enters command mode for an installed storage card.
Step 3	Server /chassis/storageadapter # **clear-foreign-config**	You are prompted to confirm the action. Enter **yes** to confirm. **Note** If you do not enter **yes**, the action is aborted.

This example shows how to clear all foreign configurations on the MegaRAID controller in slot 3:

```
Server# scope chassis
Server /chassis # scope storageadapter SLOT-3
Server /chassis/storageadapter # clear-foreign-config
Are you sure you want to clear all foreign configurations on this controller?
All data on the drive(s) will be lost.
Enter 'yes' to confirm -> yes
Server /chassis/storageadapter #
```

Deleting a Virtual Drive

Important This task deletes a virtual drive, including the drives that run the booted operating system. So back up any data that you want to retain before you delete a virtual drive.

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope storageadapter** *slot*	Enters command mode for an installed storage card.
Step 3	Server /chassis/storageadapter # **scope virtual-drive** *drive-number*	Enters command mode for the specified virtual drive.
Step 4	Server /chassis/storageadapter/virtual-drive # **delete-virtual-drive**	You are prompted to confirm the action. Enter **yes** to confirm. **Note** If you do not enter **yes**, the action is aborted.

This example shows how to delete virtual drive 3.

```
Server# scope chassis
Server /chassis # scope storageadapter SLOT-3
Server /chassis/storageadapter # scope virtual-drive 3
Server /chassis/storageadapter/virtual-drive # delete-virtual-drive
Are you sure you want to delete virtual drive 3?
All data on the drive will be lost.  Enter 'yes' to confirm -> yes
Server /chassis/storageadapter/virtual-drive #
```

Initializing a Virtual Drive

All data on a virtual drive is lost when you initialize the drive. Before you run an initialization, back up any data on the virtual drive that you want to save.

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope storageadapter** *slot*	Enters command mode for an installed storage card.
Step 3	Server /chassis/storageadapter # **scope virtual-drive** *drive-number*	Enters command mode for the specified virtual drive.
Step 4	Server /chassis/storageadapter/virtual-drive # **start-initialization**	Initializes the specified virtual drive.
Step 5	Server /chassis/storageadapter/virtual-drive # **cancel-initialization**	(Optional) Cancels the initialization of the specified virtual drive.
Step 6	Server /chassis/storageadapter/physical-drive # **get-operation-status**	Displays the status of the task that is in progress on the drive.

This example shows how to initialize virtual drive 3 using fast initialization:

```
Server# scope chassis
Server /chassis # scope storageadapter SLOT-3
Server /chassis/storageadapter # scope virtual-drive 3
Server /chassis/storageadapter/virtual-drive # start-initialization
Are you sure you want to initialize virtual drive 3?
All data on the drive will be lost.  Enter 'yes' to confirm -> yes
Fast (0) or full (1) initialization?  -> 0
Server /chassis/storageadapter/virtual-drive # get-operation-status

progress-percent: 20%
elapsed -seconds: 30
operation-in-progress: initializing virtual drive

Server /chassis/storageadapter/virtual-drive #
```

Set as Boot Drive

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope storageadapter** *slot*	Enters command mode for an installed storage card.

	Command or Action	Purpose
Step 3	Server /chassis/storageadapter # **scope virtual-drive** *drive-number*	Enters command mode for the specified virtual drive.
Step 4	Server /chassis/storageadapter # **set-boot-drive**	Specifies the controller to boot from this virtual drive.

This example shows how to specify the controller to boot from virtual drive 3:

```
Server# scope chassis
Server /chassis # scope storageadapter SLOT-3
Server /chassis/storageadapter # scope virtual-drive 3
Server /chassis/storageadapter/virtual-drive # set-boot-drive
Are you sure you want to set virtual drive 3 as the boot drive?
Enter 'yes' to confirm -> yes
Server /chassis/storageadapter/virtual-drive #
```

Modifying Attributes of a Virtual Drive

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope storageadapter** *slot*	Enters command mode for an installed storage card.
Step 3	Server /chassis/storageadapter # **scope virtual-drive** *3*	Enters the command mode for the virtual drive.
Step 4	Server /chassis/storageadapter/virtual-drive # **modify-attributes**	Prompts you to select a different current policy.

This example shows how to carve a new virtual drive out of unused space in an existing RAID 1 drive group:

```
Server# scope chassis
Server /chassis # scope storageadapter SLOT-3
Server /chassis/storageadapter # scope virtual-drive
Server /chassis/storageadapter/virtual-drive # modify-attributes

Current write policy: Write Back

  0: Write Through
  1: Write Back
  2: Write Back even if Bad BBU

Choose number from above options --> 0
```

```
The following attribute will be modified:
 - Write policy: Write Through
OK? (y or n) --> y
operation in progress.
Server /chassis/storageadapter/virtual-drive #
```

Making a Dedicated Hot Spare

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope storageadapter** *slot*	Enters command mode for an installed storage card.
Step 3	Server /chassis/storageadapter # **scope physical-drive** *drive-number*	Enters command mode for the specified physical drive.
Step 4	Server /chassis/storageadapter/physical-drive # **make-dedicated-hot-spare**	You are prompted to choose a virtual drive for which the dedicated hot spare is being created.

This example shows how to make physical drive 3 a dedicated hot spare for virtual drive 6:

```
Server# scope chassis
Server /chassis # scope storageadapter SLOT-3
Server /chassis/storageadapter # scope physical-drive 3
Server /chassis/storageadapter/physical-drive # make-dedicated-hot-spare
  5: VD_OS_1, RAID 0, 102400 MB, physical disks: 1
  6: VD_OS_2, RAID 0, 12288 MB, physical disks: 1
  7: VD_OS_3, RAID 0, 12288 MB, physical disks: 1
  8: VD_DATA_1, RAID 0, 12512 MB, physical disks: 1
  9: RAID1_2358, RAID 1, 40000 MB, physical disks: 2,3,5,8
  11: JFB_RAID1_67, RAID 1, 20000 MB, physical disks: 6,7
  12: JFB_Crv_R1_40, RAID 1, 40000 MB, physical disks: 6,7
  13: JFB_R1_10GB, RAID 1, 10000 MB, physical disks: 6,7

  Please choose from the above 8 virtual drives-->6

Server /chassis/storageadapter/physical-drive #
```

Making a Global Hot Spare

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope storageadapter** *slot*	Enters command mode for an installed storage card.
Step 3	Server /chassis/storageadapter # **scope physical-drive** *drive-number*	Enters command mode for the specified physical drive.
Step 4	Server /chassis/storageadapter/physical-drive # **make-global-hot-spare**	
Step 5	Server /chassis/storageadapter/physical-drive # **get-operation-status**	Displays the status of the task that is in progress on the drive.

This example shows how to make physical drive 3 a global hot spare:

```
Server# scope chassis
Server /chassis # scope storageadapter SLOT-3
Server /chassis/storageadapter # scope physical-drive 3
Server /chassis/storageadapter/physical-drive # make-global-hot-spare
Server /chassis/storageadapter/physical-drive #
```

Preparing a Drive for Removal

You can confirm this task only on physical drives that display the **Unconfigured Good** status.

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope storageadapter** *slot*	Enters command mode for an installed storage card.
Step 3	Server /chassis/storageadapter # **scope physical-drive** *drive-number*	Enters command mode for the specified physical drive.
Step 4	Server /chassis/storageadapter/physical-drive # **prepare-for-removal**	

This example shows how to prepare physical drive 3 for removal.

```
Server# scope chassis
Server /chassis # scope storageadapter SLOT-3
```

```
Server /chassis/storageadapter # scope physical-drive 3
Server /chassis/storageadapter/physical-drive # prepare-for-removal
Server /chassis/storageadapter/physical-drive #
```

Removing a Drive from Hot Spare Pools

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope storageadapter** *slot*	Enters command mode for an installed storage card.
Step 3	Server /chassis/storageadapter # **scope physical-drive** *drive-number*	Enters command mode for the specified physical drive.
Step 4	Server /chassis/storageadapter/physical-drive # **remove-hot-spare**	Removes a drive from the host spare pool.

This example shows how to remove physical drive 3 from the hot spare pools:

```
Server# scope chassis
Server /chassis # scope storageadapter SLOT-3
Server /chassis/storageadapter # scope physical-drive 3
Server /chassis/storageadapter/physical-drive # remove-hot-spare
Server /chassis/storageadapter/physical-drive #
```

Undo Preparing a Drive for Removal

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope storageadapter** *slot*	Enters command mode for an installed storage card.
Step 3	Server /chassis/storageadapter # **scope physical-drive** *drive-number*	Enters command mode for the specified physical drive.

	Command or Action	Purpose
Step 4	Server /chassis/storageadapter/physical-drive # **undo-prepare-for-removal**	

This example shows how to respin physical drive 3 after preparing the drive for removal.

```
Server# scope chassis
Server /chassis # scope storageadapter SLOT-3
Server /chassis/storageadapter # scope physical-drive 3
Server /chassis/storageadapter/physical-drive # undo-prepare-for-removal
Server /chassis/storageadapter/physical-drive #
```

Enabling Auto Learn Cycles for the Battery Backup Unit

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope storageadapter** *slot*	Enters command mode for an installed storage card.
Step 3	Server /chassis/storageadapter # **scope bbu**	Enter the battery backup unit command mode.
Step 4	Server /chassis/storageadapter # **enable-auto-learn**	Enables the battery auto-learn cycles

This example shows how to enable the battery auto-learn cycles:

```
Server # scope chassis
Server /chassis # scope storageadapter SLOT-2
Server /chassis/storageadapter # scope bbu
Server /chassis/storageadapter/bbu # enable-auto-learn
Automatic BBU learn cycles will occur without notice if enabled.
Are you sure? [y/n] --> y
enable-auto-learn initiated
Server /chassis/storageadapter/bbu #
```

Disabling Auto Learn Cycles for the Battery Backup Unit

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope storageadapter** *slot*	Enters command mode for an installed storage card.
Step 3	Server /chassis/storageadapter # **scope bbu**	Enter the battery backup unit command mode.
Step 4	Server /chassis/storageadapter # **disable-auto-learn**	Disables the battery auto-learn cycles

This example shows how to disables the battery auto-learn cycles:

```
Server # scope chassis
Server /chassis # scope storageadapter SLOT-2
Server /chassis/storageadapter # scope bbu
Server /chassis/storageadapter/bbu # disable-auto-learn
Automatic BBU learn cycles will no longer occur if disabled.
Are you sure? [y/n] --> y
disable-auto-learn initiated

Server /chassis/storageadapter/bbu #
```

Starting a Learn Cycle for a Battery Backup Unit

Before You Begin

You must be logged in as an admin to use this command.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope storageadapter** *slot*	Enters command mode for an installed storage card.
Step 3	Server /chassis/storageadapter # **scope bbu**	Enter the battery backup unit command mode.
Step 4	Server /chassis/storageadapter # **start-learn-cycle**	Starts the learn cycle for the battery.

This example shows how to initiate the learn cycles for a battery:

```
Server # scope chassis
Server /chassis # scope storageadapter SLOT-2
Server /chassis/storageadapter # scope bbu
Server /chassis/storageadapter/bbu # start-learn-cycle
Server /chassis/storageadapter/bbu #
```

Toggling the Locator LED for a Physical Drive

Before You Begin

You must be logged in as an admin to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope storageadapter** *slot*	Enters command mode for an installed storage card.
Step 3	Server /chassis/storageadapter # **scope physical-drive 3**	Enters the physical drive command mode.
Step 4	Server /chassis/storageadapter/physical-drive # **locator-led** {**on** | **off**}	Enables or disables the physical drive locator LED.

This example shows how to enable the locator LED for physical drive 3:

```
Server # scope chassis
Server /chassis # scope storageadapter SLOT-2
Server /chassis/storageadapter # scope physical-drive 3
Server /chassis/storageadapter/physical-drive # locator-led on
Server /chassis/storageadapter/physical-drive* # commit
Server /chassis/storageadapter/physical-drive #
```

Viewing Storage Controller Logs

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope storageadapter** *slot*	Enters command mode for an installed storage card.
Step 3	Server /chassis/storageadapter # **show log**	Displays the storage controller logs.

This example shows how to display storage controller logs:

```
Server # scope chassis
Server /chassis # scope storageadapter SLOT-3
```

```
Server /chassis/storageadapter # show log

Time                        Severity         Description
----                        --------        -------------
Fri March 1 09:52:19 2013   Warning      Predictive Failure
Fri March 1 07:50:19 2013   Info         Battery charge complete
Fri March 1 07:50:19 2013   Info         Battery charge started
Fri March 1 07:48:19 2013   Info         Battery relearn complete
Fri March 1 07:47:19 2013   Info         Battery is discharging
Fri March 1 07:45:19 2013   Info         Battery relearn started

Server /chassis/storageadapter #
```

Backing Up and Restoring the Adapter Configuration

Exporting the Adapter Configuration

The adapter configuration can be exported as an XML file to a TFTP server.

Important If any firmware or BIOS updates are in progress, do not export the adapter configuration until those tasks are complete.

Before You Begin

A supported Virtual Interface Card (VIC) must be installed in the chassis and the server must be powered on.

Obtain the TFTP server IP address.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** The server must be powered on before you can view or change adapter settings.
Step 3	Server /chassis/adapter # **export-vnic** *protocol remote server IP address*	Starts the export operation. The adapter configuration file will be stored at the specified path and filename on the remote server at the specified IP address. The protocol can be one of the following: • TFTP • FTP • SFTP • SCP • HTTP

	Command or Action	Purpose

This example exports the configuration of adapter 1:

```
Server# scope chassis
Server /chassis # scope adapter 1
Server /chassis/adapter # export-vnic ftp 192.0.20.34 //test/dnld-ucs-k9-bundle.1.0.2h.bin
Server /chassis/adapter #
```

Importing the Adapter Configuration

Important If any firmware or BIOS updates are in progress, do not import the adapter configuration until those tasks are complete.

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **scope adapter** *index*	Enters the command mode for the adapter card at the PCI slot number specified by *index*. **Note** The server must be powered on before you can view or change adapter settings.
Step 3	Server /chassis/adapter # **import-vnic** *tftp-ip-address path-and-filename*	Starts the import operation. The adapter downloads the configuration file from the specified path on the TFTP server at the specified IP address. The configuration will be installed during the next server reboot.

This example imports a configuration for the adapter in PCI slot 1:

```
Server# scope chassis
Server /chassis # scope adapter 1
Server /chassis/adapter # import-vnic 192.0.2.34 /ucs/backups/adapter4.xml
Import succeeded.
New VNIC adapter settings will take effect upon the next server reset.
Server /chassis/adapter #
```

What to Do Next

Reboot the server to apply the imported configuration.

Restoring Adapter Defaults

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **adapter-reset-defaults** *index*	Restores factory default settings for the adapter at the PCI slot number specified by the *index* argument.

This example restores the default configuration of the adapter in PCI slot 1:

```
Server# scope chassis
Server /chassis # adapter-reset-defaults 1
This operation will reset the adapter to factory default.
All your configuration will be lost.
Continue?[y|N] y
Server /chassis #
```

Managing Adapter Firmware

Adapter Firmware

A Cisco UCS C-Series network adapter contains the following firmware components:

- Adapter firmware—The main operating firmware, consisting of an active and a backup image, can be installed from the CIMC GUI or CLI interface or from the Host Upgrade Utility (HUU). You can upload a firmware image from either a local file system or a TFTP server.
- Bootloader firmware—The bootloader firmware cannot be installed from the CIMC GUI or CLI. You can install this firmware using the Host Upgrade Utility.

Installing Adapter Firmware

Important If any firmware or BIOS updates are in progress, do not install the adapter firmware until those tasks are complete.

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server /chassis # **update-adapter-fw** *tftp-ip-address path-and-filename* {**activate** | **no-activate**} [*pci-slot*] [*pci-slot*]	Downloads the specified adapter firmware file from the TFTP server, then installs the firmware as the backup image on one or two specified adapters or, if no adapter is specified, on all adapters. If the **activate** keyword is specified, the new firmware is activated after installation.
Step 3	Server /chassis # **recover-adapter-update** [*pci-slot*] [*pci-slot*]	(Optional) Clears an incomplete firmware update condition on one or two specified adapters or, if no adapter is specified, on all adapters.

This example begins an adapter firmware upgrade on the adapter in PCI slot 1:

```
Server# scope chassis
Server /chassis # update-adapter-fw 192.0.2.34 /ucs/adapters/adapter4.bin activate 1
Server /chassis #
```

What to Do Next

To activate the new firmware, see Activating Adapter Firmware, on page 55.

Activating Adapter Firmware

Important While the activation is in progress, do not:

- Reset, power off, or shut down the server.
- Reboot or reset CIMC.
- Activate any other firmware.
- Export technical support or configuration data.

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.

	Command or Action	Purpose
Step 2	Server /chassis # **activate-adapter-fw** *pci-slot* {**1** | **2**}	Activates adapter firmware image 1 or 2 on the adapter in the specified PCI slot. **Note** The changes will take effect upon the next server reboot.

This example activates adapter firmware image 2 on the adapter in PCI slot 1:

```
Server# scope chassis
Server /chassis # activate-adapter-fw 1 2
Firmware image activation suceeded
Please reset the server to run the activated image
Server /chassis #
```

What to Do Next

Reboot the server to apply the changes.

Resetting the Adapter

Before You Begin

You must log in with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope chassis**	Enters the chassis command mode.
Step 2	Server/chassis # **adapter-reset** *index*	Resets the adapter at the PCI slot number specified by the *index* argument. **Note** Resetting the adapter also resets the host.

This example resets the adapter in PCI slot 1:

```
Server# scope chassis
Server /chassis # adapter-reset 1
This operation will reset the adapter and the host if it is on.
You may lose connectivity to the CIMC and may have to log in again.
Continue?[y|N] y
Server /chassis #
```