



Managing Network Adapters

This chapter includes the following sections:

- [Overview of the Cisco UCS C-Series Network Adapters, page 1](#)
- [Viewing Network Adapter Properties, page 4](#)
- [Configuring Network Adapter Properties, page 5](#)
- [Managing vHBAs, page 6](#)
- [Managing vNICs, page 18](#)
- [Managing VM FEX, page 37](#)
- [Backing Up and Restoring the Adapter Configuration, page 42](#)
- [Managing Adapter Firmware, page 45](#)

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 VIC 1225 Virtual Interface Card
- Cisco UCS VIC 1385 Virtual Interface Card
- Cisco UCS VIC 1227T Virtual Interface Card
- Cisco UCS VIC 1387 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 policies 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 VIC 1225 Virtual Interface Card

The Cisco UCS VIC 1225 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.

Cisco UCS VIC 1385 Virtual Interface Card

The Cisco UCS VIC 1385 Virtual Interface Card is a dual-port Enhanced Quad Small Form-Factor Pluggable (QSFP) 40 Gigabit Ethernet and Fibre Channel over Ethernet (FCoE)-capable half-height PCI Express (PCIe) card designed exclusively for Cisco UCS C-Series Rack Servers. It incorporates Cisco's next-generation converged network adapter (CNA) technology, with a comprehensive feature set, providing investment protection for future feature software releases. The card enables a policy-based, stateless, agile server infrastructure that can present over 256 PCIe standards-compliant interfaces to the host that can be dynamically configured as either network interface cards (NICs) or host bus adapters (HBAs). In addition, the Cisco UCS

VIC 1385 card supports Cisco Data Center Virtual Machine Fabric Extender (VM-FEX) technology, which extends the Cisco UCS fabric interconnect ports to virtual machines, simplifying server virtualization deployment.

The personality of the card is determined dynamically at boot time using the service profile associated with the server. The number, type (NIC or HBA), identity (MAC address and World Wide Name [WWN]), failover policy, bandwidth, and quality-of-service (QoS) policies of the PCIe interfaces are all determined using the service profile. The capability to define, create, and use interfaces on demand provides a stateless and agile server infrastructure. Additional features and capabilities include:

- Each PCIe interface created on the VIC is associated with an interface on the Cisco UCS fabric interconnect, providing complete network separation for each virtual cable between a PCIe device on the VIC and the interface on the fabric interconnect
- The Cisco UCS VIC 1385 Virtual Interface Card provides high network performance and low latency for the most demanding applications such as SMB-Direct, VMQ, DPDK, and Cisco NetFlow

Cisco UCS VIC 1227T Virtual Interface Card

The Cisco UCS VIC 1227T Virtual Interface Card is a dual-port 10GBASE-T (RJ-45) 10-Gbps Ethernet and Fibre Channel over Ethernet (FCoE)-capable PCI Express (PCIe) modular LAN-on-motherboard (mLOM) adapter designed exclusively for Cisco UCS C-Series Rack Servers. New to Cisco rack servers, the mLOM slot can be used to install a Cisco VIC without consuming a PCIe slot, which provides greater I/O expandability. It incorporates next-generation converged network adapter (CNA) technology from Cisco, providing Fibre Channel connectivity over low-cost twisted pair cabling with a bit error rate (BER) of 10 to 15 up to 30 meters and investment protection for future feature releases. The mLOM card enables a policy-based, stateless, agile server infrastructure that can present up to 256 PCIe standards-compliant interfaces to the host that can be dynamically configured as either network interface cards (NICs) or host bus adapters (HBAs). In addition, the Cisco UCS VIC 1227T Virtual Interface Card supports Cisco Data Center Virtual Machine Fabric Extender (VM-FEX) technology, which extends the Cisco UCS fabric interconnect ports to virtual machines, simplifying server virtualization deployment. Additional features and capabilities include:

- Stateless and agile design - The personality of the card is determined dynamically at boot time using the service profile associated with the server. The number, type (NIC or HBA), identity (MAC address and World Wide Name [WWN]), failover policy, bandwidth, and quality-of-service (QoS) policies of the PCIe interfaces are all determined using the service profile. The capability to define, create, and use interfaces on demand provides a stateless and agile server infrastructure.
- Each PCIe interface created on the VIC is associated with an interface on the Cisco UCS fabric interconnect, providing complete network separation for each virtual cable between a PCIe device on the VIC and the interface on the fabric interconnect.
- Cisco SingleConnect technology provides an exceptionally easy, intelligent, and efficient way to connect and manage computing in the data center. Cisco SingleConnect technology dramatically simplifies the way that data centers connect to rack and blade servers, physical servers, virtual machines, LANs, SANs, and management networks.

Cisco UCS VIC 1387 Virtual Interface Card

The Cisco UCS VIC 1387 Virtual Interface Card is a dual-port Enhanced Quad Small Form-Factor Pluggable (QSFP) 40 Gigabit Ethernet and Fibre Channel over Ethernet (FCoE)-capable half-height PCI Express (PCIe) card designed exclusively for Cisco UCS C-Series Rack Servers. It incorporates Cisco's next-generation converged network adapter (CNA) technology, with a comprehensive feature set, providing investment protection for future feature software releases. The card enables a policy-based, stateless, agile server

infrastructure that can present over 256 PCIe standards-compliant interfaces to the host that can be dynamically configured as either network interface cards (NICs) or host bus adapters (HBAs). In addition, the Cisco UCS VIC 1387 card supports Cisco Data Center Virtual Machine Fabric Extender (VM-FEX) technology, which extends the Cisco UCS fabric interconnect ports to virtual machines, simplifying server virtualization deployment.

The personality of the card is determined dynamically at boot time using the service profile associated with the server. The number, type (NIC or HBA), identity (MAC address and World Wide Name [WWN]), failover policy, bandwidth, and quality-of-service (QoS) policies of the PCIe interfaces are all determined using the service profile. The capability to define, create, and use interfaces on demand provides a stateless and agile server infrastructure. Additional features and capabilities include:

- Each PCIe interface created on the VIC is associated with an interface on the Cisco UCS fabric interconnect, providing complete network separation for each virtual cable between a PCIe device on the VIC and the interface on the fabric interconnect
- The Cisco UCS VIC 1387 Virtual Interface Card provides high network performance and low latency for the most demanding applications such as SMB-Direct, VMQ, DPDK, and Cisco NetFlow

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 [<i>index</i>] [detail]	Displays adapter properties. To display the properties of a single adapter, specify the PCI slot number as the <i>index</i> 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 <i>index</i>	Enters the command mode for the adapter card at the PCI slot number specified by <i>index</i> . 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 lldp {disable enable}	Enables or disables Link Layer Discovery Protocol (LLDP) on the adapter card. LLDP is enabled by default. Note We recommend that you do not disable LLDP option, as it disables all the Data Center Bridging Capability Exchange protocol (DCBX) functionality.
Step 6	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: <ul style="list-style-type: none"> • 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

	Command or Action	Purpose
Step 7	Server /chassis/adapter # configure-vmfex <i>port-count</i>	If NIV mode is enabled, <i>port-count</i> specifies the number of VM FEX interfaces you want Cisco IMC to create, from 0 to 112.
Step 8	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 VIC 1225 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 VIC 1225 Virtual Interface Card in an FCoE application, you must associate the vHBA with the FCoE VLAN. Follow the instructions in the **Modifying vHBA Properties** section 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 <i>index</i>	Enters the command mode for the adapter card at the PCI slot number specified by <i>index</i> . Note The server must be powered on before you can view or change adapter settings.

	Command or Action	Purpose
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 <i>index</i>	Enters the command mode for the adapter card at the PCI slot number specified by <i>index</i> . Note The server must be powered on before you can view or change adapter settings.

	Command or Action	Purpose
Step 4	Server /chassis/adapter # scope host-fc-if { fc0 fc1 <i>name</i> }	Enters the host Fibre Channel interface command mode for the specified vHBA.
Step 5	Server /chassis/adapter/host-fc-if # set wwnn <i>wwnn</i>	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 <i>wwpn</i>	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 <i>mac-addr</i>	Specifies a MAC address for the vHBA.
Step 10	Server /chassis/adapter/host-fc-if # set vlan { none <i>vlan-id</i> }	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 <i>cos-value</i>	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.
Step 12	Server /chassis/adapter/host-fc-if # set rate-limit { off <i>rate</i> }	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 <i>0-99</i> }	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 <i>msec</i>	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 <i>msec</i>	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.

	Command or Action	Purpose
Step 16	Server /chassis/adapter/host-fc-if # set max-field-size <i>size</i>	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 <i>msec</i>	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 <i>count</i>	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 <i>msec</i>	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.
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: <ul style="list-style-type: none"> • 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.

	Command or Action	Purpose
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-logic	Enters the Fibre Channel fabric login command mode.
Step 31	Server /chassis/adapter/host-fc-if/port-f-logic # 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-logic # 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-logic # exit	Exits to the host Fibre Channel interface command mode.
Step 34	Server /chassis/adapter/host-fc-if# scope port-p-logic	Enters the Fibre Channel port login command mode.
Step 35	Server /chassis/adapter/host-fc-if/port-p-logic # 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-logic # 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-logic # exit	Exits to the host Fibre Channel interface command mode.

	Command or Action	Purpose
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.
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 <i>index</i>	Enters the command mode for the adapter card at the PCI slot number specified by <i>index</i> . Note The server must be powered on before you can view or change adapter settings.
Step 3	Server /chassis/adapter # create host-fc-if <i>name</i>	Creates a vHBA and enters the host Fibre Channel interface command mode. The <i>name</i> argument can be up to 32 ASCII characters.
Step 4	Server /chassis/adapter/host-fc-if # set channel-number <i>number</i>	(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 7.

Deleting a vHBA

Procedure

	Command or Action	Purpose
Step 1	Server# scope chassis	Enters the chassis command mode.
Step 2	Server /chassis # scope adapter <i>index</i>	Enters the command mode for the adapter card at the PCI slot number specified by <i>index</i> . Note The server must be powered on before you can view or change adapter settings.
Step 3	Server /chassis/adapter # delete host-fc-if <i>name</i>	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 <i>index</i>	Enters the command mode for the adapter card at the PCI slot number specified by <i>index</i> . 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 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 <i>index</i> . 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. <ul style="list-style-type: none"> • <i>wwpn</i> — The World Wide Port Name (WWPN) for the boot target in the form hh:hh:hh:hh:hh:hh:hh:hh. • <i>lun-id</i> —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 <i>index</i>	Enters the command mode for the adapter card at the PCI slot number specified by <i>index</i> . Note The server must be powered on before you can view or change adapter settings.
Step 3	Server /chassis/adapter # scope host-fc-if { <i>fc0</i> <i>fc1</i> <i>name</i> }	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 <i>entry</i>	Deletes the boot table entry at the specified position in the table. The range of <i>entry</i> 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 <i>index</i>	Enters the command mode for the adapter card at the PCI slot number specified by <i>index</i> . Note The server must be powered on before you can view or change adapter settings.
Step 3	Server /chassis/adapter # scope host-fc-if { <i>fc0</i> <i>fc1</i> <i>name</i> }	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 <i>index</i>	Enters the command mode for the adapter card at the PCI slot number specified by <i>index</i> . 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 <i>name</i> }	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 <i>index</i>	Enters the command mode for the adapter card at the PCI slot number specified by <i>index</i> . 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 # 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 VIC 1225 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.

Cisco C-series servers use Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) for packet transfers. RoCE defines the mechanism of performing RDMA over ethernet, based on the similar mechanism of RDMA over Infiniband. However, RoCE, with its performance oriented characteristics, delivers a superior performance compared to traditional network socket implementation because of the lower latency, lower CPU utilization and higher utilization of network bandwidth. RoCE meets the requirement of moving large amount of data across networks very efficiently.

The RoCE firmware requires the following configuration parameters provided by Cisco UCS Manager for better vNIC performance:

- Queue Pairs
- Memory Regions
- Resource Groups

Viewing vNIC Properties

Procedure

	Command or Action	Purpose
Step 1	Server# scope chassis	Enters the chassis command mode.
Step 2	Server /chassis # scope adapter <i>index</i>	Enters the command mode for the adapter card at the PCI slot number specified by <i>index</i> . 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 <i>name</i>] [detail]	Displays properties of a single vNIC, if specified, or all vNICs.
Step 4	Server /chassis/adapter # show ext-eth-if [detail]	Displays the external ethernet interfaces' details.

Following examples display the brief properties of all vNICs and the detailed properties of eth0 and the external interfaces:

```
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  iSCSI  Boot  usNIC
-----
eth0  1500  0             74:A2:E6:28:C6:AE  N/A  N/A  disabled  disabled  0
eth1  1500  1             74:A2:E6:28:C6:AF  N/A  N/A  disabled  disabled  0
srg   1500  0             74:A2:E6:28:C6:B2  N/A  N/A  disabled  disabled  64
hhh   1500  0             74:A2:E6:28:C6:B3  N/A  N/A  disabled  disabled  0
```

```
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 Link: 0
  PCI Order: ANY
  VLAN: NONE
  VLAN Mode: TRUNK
  Rate Limiting: OFF
  PXE Boot: disabled
  iSCSI Boot: disabled
  usNIC: 0
  Channel Number: N/A
  Port Profile: N/A
  Uplink Failover: disabled
  Uplink Failback Timeout: 5
  aRFS: disabled
  VMQ: disabled
  NVGRE: disabled
  VXLAN: disabled
  RDMA Queue Pairs: 1
  RDMA Memory Regions: 4096
  RDMA Resource Groups: 1
  CDN Name: VIC-1-eth0
```

```
Server# scope chassis
```

```

Server /chassis # scope adapter 1
Server /chassis/adapter # show ext-eth-if
Port MAC Address          Link State Encap.. Mode Admin Speed Oper..Speed  Link Training
Connector Present Connector Supported
-----
0 74:A2:E6:28:C6:A2 Link      CE          40Gbps      40Gbps      N/A
Yes Yes
1 74:A2:E6:28:C6:A3 Link      CE          40Gbps      40Gbps      N/A
Yes Yes

Server /chassis/adapter # show ext-eth-if detail

C220-FCH1834V23X /chassis/adapter # show ext-eth-if detail
Port 0:
  MAC Address: 74:A2:E6:28:C6:A2
  Link State: Link
  Encapsulation Mode: CE
  Admin Speed: 40Gbps
  Operating Speed: 40Gbps
  Link Training: N/A
  Connector Present: Yes
  Connector Supported: Yes
  Connector Type: QSFP_XCVR_CR4
  Connector Vendor: CISCO
  Connector Part Number: 2231254-3
  Connector Part Revision: B
Port 1:
  MAC Address: 74:A2:E6:28:C6:A3
  Link State: Link
  Encapsulation Mode: CE
  Admin Speed: 40Gbps
  Operating Speed: 40Gbps
  Link Training: N/A
  Connector Present: Yes
  Connector Supported: Yes
  Connector Type: QSFP_XCVR_CR4
  Connector Vendor: CISCO
  Connector Part Number: 2231254-3
  Connector Part Revision: B

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 <i>index</i>	Enters the command mode for the adapter card at the PCI slot number specified by <i>index</i> .
		Note The server must be powered on before you can view or change adapter settings.

	Command or Action	Purpose
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. 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: <ul style="list-style-type: none"> • 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: <ul style="list-style-type: none"> • access —The vNIC belongs to only one VLAN. When the VLAN is set to access mode, any frame received from the specified default VLAN (1-4094) that is received from the switch with a TAG removes that TAG when it is sent to the host OS through the vNIC. • 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.

	Command or Action	Purpose
Step 13	Server /chassis/adapter/host-eth-if # set rate-limit { <i>off</i> <i>rate</i> }	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 { <i>disable</i> <i>enable</i> }	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 <i>number</i>	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 <i>name</i>	If NIV mode is enabled for the adapter, select the port profile that should be associated with the vNIC. Note The <i>name</i> 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 { <i>disable</i> <i>enable</i> }	If NIV mode is enabled for the adapter, enable this setting if traffic on this vNIC should fail over to the secondary interface if there are communication problems.
Step 18	Server /chassis/adapter/host-eth-if # set uplink-failback-timeout <i>seconds</i>	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 <i>seconds</i> between 0 and 600.
Step 19	Server /chassis/adapter/host-eth-if # set vmq { <i>disable</i> <i>enable</i> }	Enables or disables Virtual Machine Queue (VMQ) for this adapter. Note Ensure that VMQ is not enabled when SR-IOV or netflow is enabled on the adapter.
Step 20	Server /chassis/adapter/host-eth-if # set arfs { <i>disable</i> <i>enable</i> }	Enables or disables Accelerated Receive Flow steering (aRFS) for this adapter.
Step 21	Server /chassis/adapter/host-eth-if # scope interrupt	Enters the interrupt command mode.
Step 22	Server /chassis/adapter/host-eth-if/interrupt # set interrupt-count <i>count</i>	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 23	Server /chassis/adapter/host-eth-if/interrupt # set coalescing-time <i>usec</i>	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).

	Command or Action	Purpose
Step 24	Server /chassis/adapter/host-eth-if/interrupt # set coalescing-type {idle min}	The coalescing types are as follows: <ul style="list-style-type: none"> • idle —The system does not send an interrupt until there is a period of no activity lasting 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 25	Server /chassis/adapter/host-eth-if/interrupt # set interrupt-mode {intx msi msix}	Specifies the Ethernet interrupt mode. The modes are as follows: <ul style="list-style-type: none"> • 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 26	Server /chassis/adapter/host-eth-if/interrupt # exit	Exits to the host Ethernet interface command mode.
Step 27	Server /chassis/adapter/host-eth-if # scope rcv-queue	Enters receive queue command mode.
Step 28	Server /chassis/adapter/host-eth-if/rcv-queue # set rq-count count	The number of receive queue resources to allocate. The range is 1 to 256; the default is 4.
Step 29	Server /chassis/adapter/host-eth-if/rcv-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 30	Server /chassis/adapter/host-eth-if/rcv-queue # exit	Exits to the host Ethernet interface command mode.
Step 31	Server /chassis/adapter/host-eth-if # scope trans-queue	Enters transmit queue command mode.
Step 32	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 33	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.

	Command or Action	Purpose
Step 34	Server /chassis/adapter/host-eth-if/trans-queue # exit	Exits to the host Ethernet interface command mode.
Step 35	Server /chassis/adapter/host-eth-if # scope comp-queue	Enters completion queue command mode.
Step 36	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 37	Server /chassis/adapter/host-eth-if/comp-queue # exit	Exits to the host Ethernet interface command mode.
Step 38	Server /chassis/adapter/host-eth-if/ # set rdma_mrnumber	Sets the number of memory regions to be used per adapter. The values range from 4096 to 524288.
Step 39	Server /chassis/adapter/host-eth-if/ # set rdma_qpnumber	Sets the number of queue pairs to be used per adapter. The values range from 1-8192 queue pairs.
Step 40	Server /chassis/adapter/host-eth-if/ # set rdma_resgrpnumber	Sets the number of resource groups to be used. The values range from 1-128 resource groups. Note After committing the RoCE details, you are required to reboot the server for the changes to take place.
Step 41	Server /chassis/adapter/host-eth-if # scope offload	Enters TCP offload command mode.
Step 42	Server /chassis/adapter/host-eth-if/offload # set tcp-segment-offload {disable enable}	Enables or disables TCP Segmentation Offload as follows: <ul style="list-style-type: none"> • 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).
Step 43	Server /chassis/adapter/host-eth-if/offload # set tcp-rx-checksum-offload {disable enable}	Enables or disables TCP Receive Offload Checksum Validation as follows: <ul style="list-style-type: none"> • 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.

	Command or Action	Purpose
Step 44	Server /chassis/adapter/host-eth-if/offload # set tcp-tx-checksum-offload {disable enable}	Enables or disables TCP Transmit Offload Checksum Validation as follows: <ul style="list-style-type: none"> • 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 45	Server /chassis/adapter/host-eth-if/offload # set tcp-large-receive-offload {disable enable}	Enables or disables TCP Large Packet Receive Offload as follows: <ul style="list-style-type: none"> • 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 46	Server /chassis/adapter/host-eth-if/offload # exit	Exits to the host Ethernet interface command mode.
Step 47	Server /chassis/adapter/host-eth-if # scope rss	Enters Receive-side Scaling (RSS) command mode.
Step 48	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 49	Server /chassis/adapter/host-eth-if/rss # set rss-hash-ipv4 {disable enable}	Enables or disables IPv4 RSS. The default is enable.
Step 50	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 51	Server /chassis/adapter/host-eth-if/rss # set rss-hash-ipv6 {disable enable}	Enables or disables IPv6 RSS. The default is enable.
Step 52	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 53	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 54	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.

	Command or Action	Purpose
Step 55	Server /chassis/adapter/host-eth-if/rss # exit	Exits to the host Ethernet interface command mode.
Step 56	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 # enable vmq
Server /chassis/adapter/host-eth-if # enable arfs
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.

Enabling or Disabling Link Training on External Ethernet Interfaces

Link training for the port profile on the external ethernet interfaces of the specified vNIC can be enabled or disabled.

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 <i>index</i>	Enters the command mode for the adapter card at the PCI slot number specified by <i>index</i> . Note The server must be powered on before you can view or change adapter settings.

	Command or Action	Purpose
Step 4	Server /chassis / adapter # scope ext-eth-if 0 1 name	Enters the external ethernet interface command mode for the specified vNIC.
Step 5	Server /chassis / adapter / ext-eth-if # set link-training on off	Enables or disables the link training for the specified vNIC.
Step 6	Server /chassis / adapter / ext-eth-if * # commit	Commits the transaction to the system configuration.

This example shows how to enable or disable link training on the external ethernet interface.

```
Server# scope chassis
Server /chassis # scope adapter 1
Server /chassis/adapter # scope ext-eth-if 1
Server /chassis/adapter/ext-eth-if* # set link-training on
Server /chassis/adapter/ext-eth-if# commit
You may lose connectivity to the Cisco IMC and may have to log in again.
Do you wish to continue? [y/N] y
Port 1:
  MAC Address: 74:A2:E6:28:C6:A3
  Link State: Link
  Encapsulation Mode: CE
  Admin Speed: 40Gbps
  Operating Speed: -
  Link Training: N/A
  Connector Present: Yes
  Connector Supported: Yes
  Connector Type: QSFP_XCVR_CR4
  Connector Vendor: CISCO
  Connector Part Number: 2231254-3
  Connector Part Revision: B
```

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 <i>index</i> . Note The server must be powered on before you can view or change adapter settings.

	Command or Action	Purpose
Step 3	Server /chassis/adapter # create host-eth-if <i>name</i>	Creates a vNIC and enters the host Ethernet interface command mode. The <i>name</i> argument can be up to 32 ASCII characters.
Step 4	Server /chassis/adapter/host-eth-if # set channel-number <i>number</i>	(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 <i>index</i>	Enters the command mode for the adapter card at the PCI slot number specified by <i>index</i> . Note The server must be powered on before you can view or change adapter settings.
Step 3	Server /chassis/adapter # delete host-eth-if <i>name</i>	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. 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 Cisco IMC 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 Cisco IMC 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 <i>index</i>	Enters the command mode for the adapter card at the PCI slot number specified by <i>index</i> . 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 your 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# 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 Cisco IMC 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 value using the Cisco IMC CLI , on page 31.

	Command or Action	Purpose
Step 5	server/chassis/adapter/host-eth-if/usnic-config# set cq-count <i>count</i>	Specifies the number of completion queue resources to allocate. We recommend that you set this value to 6. 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 <i>count</i>	Specifies the number of receive queue resources to allocate. We recommend that you set this value to 6.
Step 7	server/chassis/adapter/host-eth-if/usnic-config# set tq-count <i>count</i>	Specifies the number of transmit queue resources to allocate. We recommend that you set this value to 6.
Step 8	server/chassis/adapter/host-eth-if/usnic-config# set usnic-count <i>number of usNICs</i> .	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.
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.

	Command or Action	Purpose
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 6
server /chassis/adapter/host-eth-if/usnic-config *# set rq-count 6
server /chassis/adapter/host-eth-if/usnic-config *# set tq-count 6
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# scope bios
server /bios # scope advanced
server /bios/advanced # set IntelVTD Enabled
server /bios/advanced *# set ATS Enabled*
server /bios/advanced *# set CoherencySupport Enabled
server /bios/advanced *# commit
Changes to BIOS set-up parameters will require a reboot.
Do you want to reboot the system?[y|N]y
A system reboot has been initiated.
```

Modifying a Cisco usNIC value using the Cisco IMC CLI

Before You Begin

You must log in to the Cisco IMC 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 <i>index</i>	Enters the command mode for the adapter card at the PCI slot number specified by <i>index</i> .

	Command or Action	Purpose
		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 your 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 usnic-count <i>number of usNICs</i> .	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 6	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 7	server/chassis/adapter/host-eth-if/usnic-config# exit	Exits to host Ethernet interface command mode.
Step 8	server/chassis/adapter/host-eth-if# exit	Exits to adapter interface command mode.
Step 9	server/chassis/adapter# exit	Exits to chassis interface command mode.
Step 10	server/chassis# exit	Exits to server interface command mode.

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 32
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
```

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 <i>index</i>	Enters the command mode for the adapter card at the PCI slot number specified by <i>index</i> . Note The server must be powered on before you can view or change adapter settings.
Step 3	Server /chassis/adapter # scope host-eth-if { <i>eth0</i> <i>eth1</i> <i>name</i> }	Enters the host Ethernet interface command mode for the specified vNIC.
Step 4	Server /chassis/adapter/host-eth-if # show usnic-config <i>index</i>	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 Cisco IMC CLI with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	server# scope chassis	Enters chassis command mode.

	Command or Action	Purpose
Step 2	server/chassis# scope adapter <i>index</i>	Enters the command mode for the adapter card at the PCI slot number specified by <i>index</i> . 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 your 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/chassis/host-eth-if/iscsi-boot *# commit
New host-eth-if settings will take effect upon the next adapter reboot

server/chassis/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 and Nexus 6000 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 <i>index</i>	Enters the command mode for the adapter card at the PCI slot number specified by <i>index</i> . 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 <i>name</i> }	Enters the host Ethernet interface command mode for the specified vNIC.
Step 4	Server /chassis/adapter/host-eth-if # create iscsi-boot <i>index</i>	Creates the iSCSI boot index for the vNIC. At this moment, only 0 is allowed as the index.
Step 5	Server /chassis/adapter/host-eth-if/iscsi-boot* # create iscsi-target <i>index</i>	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 <i>string</i>	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 <i>index</i>	Enters the command mode for the adapter card at the PCI slot number specified by <i>index</i> . 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 <i>name</i> }	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.

For this release, VM FEX supports the following cards and Operating systems:

Cards - Cisco UCS 1225 Virtual Interface Card

Operating Systems:

- VMware ESXi 5.1 Update 2
- VMware ESXi 5.5

VM FEX is not supported on Microsoft Hyper-V and Red Hat KVM for this release.

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 <i>index</i>	Enters the command mode for the adapter card at the PCI slot number specified by <i>index</i> . 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 39.
Step 4	Server /chassis/adapter # scope vmfex <i>name</i>	Enters the command mode for the specified VM FEX interface.

	Command or Action	Purpose
Step 5	Server /chassis/adapter/vmfex # show interrupt [detail]	Displays Ethernet interrupt settings. For field descriptions, see Ethernet Interrupt Settings , on page 40.
Step 6	Server /chassis/adapter/vmfex # show rcv-queue [detail]	Displays Ethernet receive queue settings. For field descriptions, see Ethernet Receive Queue Settings , on page 40.
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 41.
Step 8	Server /chassis/adapter/vmfex # show comp-queue [detail]	Displays completion queue settings. For field descriptions, see Completion Queue Settings , on page 41.
Step 9	Server /chassis/adapter/vmfex # show offload [detail]	Displays TCP offload settings. For field descriptions, see TCP Offload Settings , on page 41.
Step 10	Server /chassis/adapter/vmfex # show rss [detail]	Displays RSS settings. For field descriptions, see Receive Side Scaling Settings , on page 42.

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 rcv-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.
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.

Name	Description
Port Profile	If NIV mode is enabled for the adapter, the port profile associated with the VM 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 Cisco IMC 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: <ul style="list-style-type: none"> • 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: <ul style="list-style-type: none"> • 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.

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 <i>index</i> .

	Command or Action	Purpose
		<p>Note The server must be powered on before you can view or change adapter settings.</p>
Step 3	<p>Server /chassis/adapter # export-vnic protocol remote server IP address</p>	<p>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:</p> <ul style="list-style-type: none"> • TFTP • FTP • SFTP • SCP • HTTP <p>Note The Cisco UCS C-Series server now supports fingerprint confirmation of the server when you update firmware through a remote server. This option is available only if you choose SCP or SFTP as the remote server type.</p> <p>If you chose SCP or SFTP as the remote server type while performing this action, a prompt with the message Server (RSA) key fingerprint is <server_finger_print_ID> Do you wish to continue? Click y or n depending on the authenticity of the server fingerprint.</p> <p>The fingerprint is based on the host's public key and helps you to identify or verify the host you are connecting to.</p>

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 <i>index</i>	Enters the command mode for the adapter card at the PCI slot number specified by <i>index</i> . Note The server must be powered on before you can view or change adapter settings.
Step 3	Server /chassis/adapter # import-vnic <i>tftp-ip-address path-and-filename</i>	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 <i>index</i>	Restores factory default settings for the adapter at the PCI slot number specified by the <i>index</i> argument. Note Resetting the adapter to default settings sets the port speed to 4 X 10 Gbps. Choose 40 Gbps as the port speed only if you are using a 40 Gbps switch.

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 Cisco IMC 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 Cisco IMC. 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 <i>tftp-ip-address path-and-filename</i> { activate no-activate } [<i>pci-slot</i>] [<i>pci-slot</i>]	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 [<i>pci-slot</i>] [<i>pci-slot</i>]	(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 46.

Activating Adapter Firmware



Important

While the activation is in progress, do not:

- Reset, power off, or shut down the server.
- Reboot or reset Cisco IMC.
- 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.
Step 2	Server /chassis# activate-adapter-fw <i>pci-slot</i> { 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 succeeded
Please reset the server to run the activated image
Server /chassis #
```

What to Do Next

Reboot the server to apply the changes.