



Cisco UCS Manager Network Management Guide Using the CLI, Release 4.1

First Published: 2020-02-20

Americas Headquarters

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
<http://www.cisco.com>
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2020 Cisco Systems, Inc. All rights reserved.



CONTENTS

PREFACE

Preface	xiii
Audience	xiii
Conventions	xiii
Related Cisco UCS Documentation	xv
Documentation Feedback	xv

CHAPTER 1

New and Changed Information	1
New and Changed Information	1

CHAPTER 2

Overview	3
Overview	3
Cisco UCS Manager User CLI Documentation	3

CHAPTER 3

LAN Connectivity	5
Fabric Interconnect Overview	5
Uplink Connectivity	5
Downlink Connectivity	6
Configuring the Fabric Interconnects	6
Fabric Interconnect Information Policy	6
Enabling the Information Policy on the Fabric Interconnect	6
Disabling the Information Policy on the Fabric Interconnect	7
Viewing the LAN Neighbors of the Fabric Interconnect	8
Viewing the SAN Neighbors of the Fabric Interconnect	9
Viewing the LLDP Neighbors of the Fabric Interconnect	9
Fabric Evacuation	10
Stopping Traffic on a Fabric Interconnect	11

Displaying the Status of Evacuation for a Fabric Interconnect	12
Displaying the Status of Evacuation for an IOM	13
Verifying Fabric Evacuation	14
Restarting Traffic on a Fabric Interconnect	15
Fabric Interconnect Port Types	16
Fabric Interconnect Switching Modes	16
Ethernet Switching Mode	16
Configuring Ethernet Switching Mode	18
Fibre Channel Switching Mode	18
Configuring Fibre Channel Switching Mode	19
<hr/>	
CHAPTER 4	LAN Ports and Port Channels 21
Unified Ports on the Cisco UCS 6200 Series and 6324 Fabric Interconnects	21
Port Modes	22
Port Types	22
Data Traffic Interruption from Port Mode Changing	23
Guidelines for Configuring Unified Ports	23
Cautions and Guidelines for Configuring Unified Uplink Ports and Unified Storage Ports	25
Configuring the Port Mode	26
Configuring Breakout Ports	28
Port Breakout Functionality on Cisco UCS 6454 Fabric Interconnects	28
Port Breakout Functionality on Cisco UCS 64108 Fabric Interconnects	29
Port Breakout Functionality on Cisco UCS 6300 Series Fabric Interconnects	30
Configuring Multiple Breakout Ports	32
Configuring a Breakout Ethernet Uplink Port	34
Configuring a Breakout Ethernet Uplink Port Channel Member	35
Configuring Ethernet Uplink Breakout Port as a Pin Group Target	36
Configuring Breakout Appliance Ports	37
Configuring a Breakout Appliance Port Channel Member	38
Configuring Breakout FCoE Storage Ports	39
Configuring a Breakout FCoE Uplink Port	40
Configuring an FCoE Port Channel Member	41
Configuring a Breakout VLAN Member Port	42
Modifying a Breakout Port	42

Un-configuring Breakout Ports	47
Deleting Breakout Ports	48
Cisco UCS Mini Scalability Ports	50
Configuring Scalability Ports	50
Beacon LEDs for Unified Ports	51
Configuring the Beacon LEDs for Unified Ports	51
Physical and Backplane Ports	52
Displaying VIF Port Statistics Obtained From the Adaptor	52
Displaying VIF Port Statistics Obtained From the ASIC	53
Displaying VIF Ports That Correspond to NIV Ports	54
Verifying Status of Backplane Ports	55
Server Ports	57
Automatic Configuration of Fabric Interconnect Server Ports	57
Automatically Configuring Server Ports	57
Configuring a Server Port	58
Unconfiguring a Server Port	59
Uplink Ethernet Ports	59
Configuring an Uplink Ethernet Port	59
Unconfiguring an Uplink Ethernet Port	60
Configuring an Uplink Ethernet Port for Forward Error Correction	61
Appliance Ports	62
Configuring an Appliance Port	62
Assigning a Target MAC Address to an Appliance Port or Appliance Port Channel	64
Creating an Appliance Port	65
Mapping an Appliance Port to a Community VLAN	66
Unconfiguring an Appliance Port	67
Configuring an Appliance Port for Forward Error Correction	68
FCoE Uplink Ports	69
Configuring a FCoE Uplink Port	69
Unconfiguring a FCoE Uplink Port	70
Viewing FCoE Uplink Ports	71
Configuring FCoE Uplink for Forward Error Correction	72
Unified Storage Ports	73
Configuring a Unified Storage Port	73

Unified Uplink Ports	74
Configuring a Unified Uplink Port	74
FCoE and Fibre Channel Storage Ports	75
Configuring a Fibre Channel Storage or FCoE Port	75
Unconfiguring a Fibre Channel Storage or FCoE Port	76
Restoring a Fibre Channel Storage Port Back to an Uplink Fibre Channel Port	77
Uplink Ethernet Port Channels	77
Configuring an Uplink Ethernet Port Channel	78
Unconfiguring an Uplink Ethernet Port Channel	79
Adding a Member Port to an Uplink Ethernet Port Channel	79
Deleting a Member Port from an Uplink Ethernet Port Channel	80
Appliance Port Channels	81
Configuring an Appliance Port Channel	81
Unconfiguring an Appliance Port Channel	83
Enabling or Disabling an Appliance Port Channel	83
Adding a Member Port to an Appliance Port Channel	84
Deleting a Member Port from an Appliance Port Channel	85
Fibre Channel Port Channels	86
Configuring a Fibre Channel Port Channel	86
Configuring a FCoE Port Channel	88
Adding Channel Mode Active To The Upstream NPIV Fibre Channel Port Channel	88
Enabling or Disabling a Fibre Channel Port Channel	89
Adding a Member Port to a Fibre Channel Port Channel	90
Deleting a Member Port from a Fibre Channel Port Channel	91
FCoE Port Channels	92
Configuring a FCoE Port Channel	92
Adding a Member Port to a FCoE Uplink Port Channel	93
Unified Uplink Port Channel	94
Configuring a Unified Uplink Port Channel	94
Event Detection and Action	95
Policy-Based Port Error Handling	95
Creating Threshold Definition	96
Configuring Error Disable on a Fabric Interconnect Port	97
Configuring Auto Recovery on a Fabric Interconnect Port	98

Viewing the Network Interface Port Error Counters	99
Adapter Port Channels	100
Viewing Adapter Port Channels	100
Fabric Port Channels	101
Load Balancing Over Ports	101
Cabling Considerations for Fabric Port Channels	102
Configuring a Fabric Port Channel	103
Viewing Fabric Port Channels	103
Enabling or Disabling a Fabric Port Channel Member Port	104

CHAPTER 5

VLANs	107
Named VLANs	107
Private VLANs	108
VLAN Port Limitations	109
Configuring Named VLANs	111
Creating a Named VLAN Accessible to Both Fabric Interconnects (Uplink Ethernet Mode)	111
Creating a Named VLAN Accessible to Both Fabric Interconnects (Ethernet Storage Mode)	112
Creating a Named VLAN Accessible to One Fabric Interconnect (Uplink Ethernet Mode)	113
Creating a Secondary VLAN for a Private VLAN (Accessible to One Fabric Interconnect)	115
Deleting a Named VLAN	116
Configuring Private VLANs	117
Creating a Primary VLAN for a Private VLAN (Accessible to Both Fabric Interconnects)	117
Creating a Primary VLAN for a Private VLAN (Accessible to One Fabric Interconnect)	118
Creating a Secondary VLAN for a Private VLAN (Accessible to Both Fabric Interconnects)	119
Creating a Secondary VLAN for a Private VLAN (Accessible to One Fabric Interconnect)	121
Allowing PVLANS on vNICs	122
Creating a Primary VLAN for a Private VLAN on an Appliance Cloud	123
Creating a Secondary VLAN for a Private VLAN on an Appliance Cloud	124
Community VLANs	125
Creating a Community VLAN	125
Viewing Community VLANs	126
Allowing Community VLANs on vNICs	126
Allowing PVLAN on Promiscuous Access or Trunk Port	127
Deleting a Community VLAN	128

Viewing the VLAN Port Count	129
VLAN Port Count Optimization	130
Enabling Port VLAN Count Optimization	131
Disabling Port VLAN Count Optimization	131
Viewing the Port VLAN Count Optimization Groups	132
VLAN Groups	132
Creating a VLAN Group	133
Creating an Inband VLAN Group	134
Viewing VLAN Groups	135
Deleting a VLAN Group	135
Modifying the Reserved VLAN	136
VLAN Permissions	137
Creating VLAN Permissions	137
Viewing VLAN Permissions	138
Deleting a VLAN Permission	138

CHAPTER 6

LAN Pin Groups	141
LAN Pin Groups	141
Configuring a LAN Pin Group	141

CHAPTER 7

MAC Pools	143
MAC Pools	143
Creating a MAC Pool	143
Deleting a MAC Pool	145

CHAPTER 8

Quality of Service	147
Quality of Service	147
Configuring System Classes	148
System Classes	148
Configuring a System Class	149
Disabling a System Class	151
Configuring Quality of Service Policies	152
Quality of Service Policy	152
Configuring a QoS Policy	152

Deleting a QoS Policy	154
Configuring Flow Control Policies	155
Flow Control Policy	155
Configuring a Flow Control Policy	155
Deleting a Flow Control Policy	157
Configuring Slow Drain	157
QoS Slow Drain Device Detection and Mitigation	157
Configuring Slow Drain Detection	158
Configuring Slow Drain Timers	159
Displaying Slow Drain Settings	160

CHAPTER 9

Port Security 161

Port Security Overview	161
Port Security Violations	162
Guidelines for Port Security on UCS 6454 Fabric Interconnects	162
Configuring Port Security	163

CHAPTER 10

Upstream Disjoint Layer-2 Networks 165

Upstream Disjoint Layer-2 Networks	165
Guidelines for Configuring Upstream Disjoint L2 Networks	166
Upstream Disjoint L2 Networks Pinning Considerations	167
Configuring Cisco UCS for Upstream Disjoint L2 Networks	169
Assigning Ports and Port Channels to VLANs	170
Removing Ports and Port Channels from VLANs	171
Viewing Ports and Port Channels Assigned to VLANs	172

CHAPTER 11

Network-Related Policies 173

Configuring vNIC Templates	173
vNIC Template	173
Creating a vNIC Template	174
Creating vNIC Template Pairs	177
Undo vNIC Template Pairs	178
Binding a vNIC to a vNIC Template	179
Unbinding a vNIC from a vNIC Template	180

Deleting a vNIC Template	181
Configuring Adapter Policies	181
Ethernet and Fibre Channel Adapter Policies	181
Accelerated Receive Flow Steering	184
Interrupt Coalescing	185
Adaptive Interrupt Coalescing	185
RDMA Over Converged Ethernet Overview	185
Creating an Ethernet Adapter Policy	188
Configuring an Ethernet Adapter Policy to Enable eNIC Support for MRQS on Linux Operating Systems	193
Configuring an Ethernet Adapter Policy to Enable eNIC Support for RSS on VMware ESXi	194
Configuring an Ethernet Adapter Policy to Enable Stateless Offloads with NVGRE	195
Configuring an Ethernet Adapter Policy to Enable Stateless Offloads with VXLAN	196
Deleting an Ethernet Adapter Policy	198
Configuring the Default vNIC Behavior Policy	198
Default vNIC Behavior Policy	198
Configuring a Default vNIC Behavior Policy	199
Configuring LAN Connectivity Policies	199
About the LAN and SAN Connectivity Policies	199
Privileges Required for LAN and SAN Connectivity Policies	200
Interactions between Service Profiles and Connectivity Policies	200
Creating a LAN Connectivity Policy	200
Deleting a LAN Connectivity Policy	202
Creating a vNIC for a LAN Connectivity Policy	203
Deleting a vNIC from a LAN Connectivity Policy	204
Creating an iSCSI vNIC for a LAN Connectivity Policy	205
Deleting a vNIC from a LAN Connectivity Policy	206
Configuring Network Control Policies	207
Network Control Policy	207
Configuring Link Layer Discovery Protocol for Fabric Interconnect vEthernet Interfaces	208
Creating a Network Control Policy	208
Deleting a Network Control Policy	210
Configuring Multicast Policies	210
Multicast Policy	210

Creating a Multicast Policy	211
Modifying a Multicast Policy	211
Deleting a Multicast Policy	212
Configuring LACP Policies	213
LACP Policy	213
Creating a LACP Policy	213
Modifying a LACP Policy	213
Configuring UDLD Link Policies	214
Understanding UDLD	214
UDLD Configuration Guidelines	215
Creating a Link Profile	216
Creating a UDLD Link Policy	216
Modifying the UDLD System Settings	216
Assigning a Link Profile to a Port Channel Ethernet Interface	216
Assigning a Link Profile to an Uplink Ethernet Interface	217
Assigning a Link Profile to a Port Channel FCoE Interface	217
Assigning a Link Profile to an Uplink FCoE Interface	217
Configuring VMQ and VMMQ Connection Policies	218
VMQ Connection Policy	218
Creating a VMQ Connection Policy	218
Assigning VMQ Setting to a vNIC	221
Enabling VMQ and NVGRE Offloading on the same vNIC	221
VMMQ Connection Policy	222
VMMQ Guidelines	222
Creating a VMMQ Connection Policy	223
Creating a QoS Policy for VMMQ	224
Assigning a VMMQ Setting to a vNIC	225
NetQueue	225
Information About NetQueue	225
Configuring NetQueue	225



Preface

- [Audience, on page xiii](#)
- [Conventions, on page xiii](#)
- [Related Cisco UCS Documentation, on page xv](#)
- [Documentation Feedback, on page xv](#)

Audience

This guide is intended primarily for data center administrators with responsibilities and expertise in one or more of the following:

- Server administration
- Storage administration
- Network administration
- Network security

Conventions

Text Type	Indication
GUI elements	GUI elements such as tab titles, area names, and field labels appear in this font . Main titles such as window, dialog box, and wizard titles appear in this font .
Document titles	Document titles appear in <i>this font</i> .
TUI elements	In a Text-based User Interface, text the system displays appears in <i>this font</i> .
System output	Terminal sessions and information that the system displays appear in <i>this font</i> .
CLI commands	CLI command keywords appear in this font . Variables in a CLI command appear in <i>this font</i> .
[]	Elements in square brackets are optional.

Text Type	Indication
{x y z}	Required alternative keywords are grouped in braces and separated by vertical bars.
[x y z]	Optional alternative keywords are grouped in brackets and separated by vertical bars.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
< >	Nonprinting characters such as passwords are in angle brackets.
[]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.



Note Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the document.



Tip Means *the following information will help you solve a problem*. The tips information might not be troubleshooting or even an action, but could be useful information, similar to a Timesaver.



Timesaver Means *the described action saves time*. You can save time by performing the action described in the paragraph.



Caution Means *reader be careful*. In this situation, you might perform an action that could result in equipment damage or loss of data.



Warning IMPORTANT SAFETY INSTRUCTIONS

This warning symbol means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. Use the statement number provided at the end of each warning to locate its translation in the translated safety warnings that accompanied this device.

SAVE THESE INSTRUCTIONS

Related Cisco UCS Documentation

Documentation Roadmaps

For a complete list of all B-Series documentation, see the *Cisco UCS B-Series Servers Documentation Roadmap* available at the following URL: https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/overview/guide/UCS_roadmap.html

For a complete list of all C-Series documentation, see the *Cisco UCS C-Series Servers Documentation Roadmap* available at the following URL: https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/overview/guide/ucs_rack_roadmap.html.

For information on supported firmware versions and supported UCS Manager versions for the rack servers that are integrated with the UCS Manager for management, refer to [Release Bundle Contents for Cisco UCS Software](#).

Other Documentation Resources

Follow [Cisco UCS Docs on Twitter](#) to receive document update notifications.

Documentation Feedback

To provide technical feedback on this document, or to report an error or omission, please send your comments to ucs-docfeedback@external.cisco.com. We appreciate your feedback.



CHAPTER 1

New and Changed Information

- [New and Changed Information](#), on page 1

New and Changed Information

This section provides information on new features and changed behavior in Cisco UCS Manager, Release 4.1.

Table 1: New Features and Changed Behavior in Cisco UCS Manager, Release 4.1(1)

Feature	Description	Where Documented
NVMe over Fabrics (NVMeoF) using RDMA over Converged Ethernet (RoCE) v2	Cisco UCS Manager 4.1(1a) now supports configuring a RoCE v2 Ethernet adapter policy to use NVMeoF protocols with Cisco UCS VIC 1400 Series adapters on Linux.	Ethernet and Fibre Channel Adapter Policies , on page 181
New algorithm for RDMA over Converged Ethernet (RoCE) protocol.	Cisco UCS Manager 4.1(1a) now supports the RoCE v2 protocol on Cisco UCS VIC 1400 Series adapters.	RDMA Over Converged Ethernet Overview , on page 185



CHAPTER 2

Overview

- [Overview, on page 3](#)
- [Cisco UCS Manager User CLI Documentation, on page 3](#)

Overview

This guide includes the following information:

- Configure/Enable Server Ports; Configure/Enable Uplink Ports; Configure/Enable FC Ports.
- Create LAN Pin Groups
- Create VLANs and VLAN groups
- Create Server Links
- Configure QoS System Class
- Configure Global Policies
- Monitor Network Health
- Traffic Monitoring

Cisco UCS Manager User CLI Documentation

Cisco UCS Manager offers you a set of smaller, use-case based documentation described in the following table:

Guide	Description
Cisco UCS Manager Getting Started Guide	Discusses Cisco UCS architecture and Day 0 operations, including Cisco UCS Manager initial configuration, and configuration best practices.

Guide	Description
Cisco UCS Manager Administration Guide	Discusses password management, role-based access configuration, remote authentication, communication services, CIMC session management, organizations, backup and restore, scheduling options, BIOS tokens and deferred deployments.
Cisco UCS Manager Infrastructure Management Guide	Discusses physical and virtual infrastructure components used and managed by Cisco UCS Manager.
Cisco UCS Manager Firmware Management Guide	Discusses downloading and managing firmware, upgrading through Auto Install, upgrading through service profiles, directly upgrading at endpoints using firmware auto sync, managing the capability catalog, deployment scenarios, and troubleshooting.
Cisco UCS Manager Server Management Guide	Discusses the new licenses, registering Cisco UCS domains with Cisco UCS Central, power capping, server boot, server profiles and server-related policies.
Cisco UCS Manager Storage Management Guide	Discusses all aspects of storage management such as SAN and VSAN in Cisco UCS Manager.
Cisco UCS Manager Network Management Guide	Discusses all aspects of network management such as LAN and VLAN connectivity in Cisco UCS Manager.
Cisco UCS Manager System Monitoring Guide	Discusses all aspects of system and health monitoring including system statistics in Cisco UCS Manager.
Cisco UCS S3260 Server Integration with Cisco UCS Manager	Discusses all aspects of management of UCS S-Series servers that are managed through Cisco UCS Manager.



CHAPTER 3

LAN Connectivity

- [Fabric Interconnect Overview, on page 5](#)
- [Uplink Connectivity, on page 5](#)
- [Downlink Connectivity, on page 6](#)
- [Configuring the Fabric Interconnects, on page 6](#)
- [Fabric Evacuation, on page 10](#)
- [Fabric Interconnect Port Types, on page 16](#)
- [Fabric Interconnect Switching Modes, on page 16](#)

Fabric Interconnect Overview

The fabric interconnect is the core component of Cisco UCS. The Cisco UCS Fabric Interconnects provide uplink access to LAN, SAN, and out-of-band management segment. Cisco UCS infrastructure management is through the embedded management software, Cisco UCS Manager, for both hardware and software management. The Cisco UCS Fabric Interconnects are Top-of-Rack devices, and provide unified access to the Cisco UCS domain.

The Cisco UCS FIs provide network connectivity and management for the connected servers. The Cisco UCS Fabric Interconnects run the Cisco UCS Manager control software and consist of expansion modules for the Cisco UCS Manager software.

For more information about Cisco UCS Fabric Interconnects, see the *Cisco UCS Manager Getting Started Guide*.

Uplink Connectivity

Use fabric interconnect ports configured as uplink ports to connect to uplink upstream network switches. Connect these uplink ports to upstream switch ports as individual links, or as links configured as port channels. Port channel configurations provide bandwidth aggregation as well as link redundancy.

You can achieve northbound connectivity from the fabric interconnect through a standard uplink, a port channel, or a virtual port channel configuration. The port channel name and ID configured on fabric interconnect should match the name and ID configuration on the upstream Ethernet switch.

It is also possible to configure a port channel as a vPC, where port channel uplink ports from a fabric interconnect are connected to different upstream switches. After all uplink ports are configured, create a port channel for these ports.

Downlink Connectivity

Each fabric interconnect is connected to IOMs in the UCS chassis, which provides connectivity to each blade server. Internal connectivity from blade servers to IOMs is transparently provided by Cisco UCS Manager using 10BASE-KR Ethernet standard for backplane implementations, and no additional configuration is required. You must configure the connectivity between the fabric interconnect server ports and IOMs. Each IOM, when connected with the fabric interconnect server port, behaves as a line card to fabric interconnect, hence IOMs should never be cross-connected to the fabric interconnect. Each IOM is connected directly to a single fabric interconnect.

The Fabric Extender (also referred to as the IOM, or FEX) logically extends the fabric interconnects to the blade server. The best analogy is to think of it as a remote line card that's embedded in the blade server chassis, allowing connectivity to the external world. IOM settings are pushed via Cisco UCS Manager and are not managed directly. The primary functions of this module are to facilitate blade server I/O connectivity (internal and external), multiplex all I/O traffic up to the fabric interconnects, and help monitor and manage the Cisco UCS infrastructure.

Configure Fabric interconnect ports that should be connected to downlink IOM cards as server ports. Make sure there is physical connectivity between the fabric interconnect and IOMs. You must also configure the IOM ports and the global chassis discovery policy.



Note

For UCS 2200 I/O modules, you can also select the Port Channel option and all I/O module-connected server ports will be automatically added to a port channel.

Configuring the Fabric Interconnects

Fabric Interconnect Information Policy

You must configure the information policy to display the uplink switches that are connected to Cisco UCS.



Important

You must enable the information policy on the fabric interconnect to view the SAN, LAN, and LLDP neighbors of the fabric interconnect.

Enabling the Information Policy on the Fabric Interconnect



Note

By default, the information policy is disabled on the fabric interconnect.

SUMMARY STEPS

1. UCS-A # **scope system**

2. UCS-A/system # **scope info-policy**
3. (Optional) UCS-A/system/info-policy # **show**
4. UCS-A/system/info-policy # **enable**
5. UCS-A/system/info-policy* # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A # scope system	Enters system mode.
Step 2	UCS-A/system # scope info-policy	Enters the information policy state.
Step 3	(Optional) UCS-A/system/info-policy # show	Shows if the information policy is enabled or disabled.
Step 4	UCS-A/system/info-policy # enable	Enables the information policy on the fabric interconnect.
Step 5	UCS-A/system/info-policy* # commit-buffer	Enables the information policy on the fabric interconnect.

Example

The following example shows how to enable the information policy on the fabric interconnect:

```
UCS-A# scope system
UCS-A/system # scope info-policy
UCS-A/system/info-policy # show
Info Policy:
State: Disabled
UCS-A/system/info-policy # enable
UCS-A/system/info-policy* # commit-buffer
UCS-A/system/info-policy #
```

Disabling the Information Policy on the Fabric Interconnect

SUMMARY STEPS

1. UCS-A # **scope system**
2. UCS-A/system # **scope info-policy**
3. (Optional) UCS-A/system/info-policy # **show**
4. UCS-A/system/info-policy # **disable**
5. UCS-A/system/info-policy* # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A # scope system	Enters system mode.
Step 2	UCS-A/system # scope info-policy	Enters the information policy state.
Step 3	(Optional) UCS-A/system/info-policy # show	Shows if the information policy is enabled or disabled.

	Command or Action	Purpose
Step 4	UCS-A/system/info-policy # disable	Disables the information policy on the fabric interconnect.
Step 5	UCS-A/system/info-policy* # commit-buffer	Disables information policy on the fabric interconnect.

Example

The following example shows how to disable the information policy on the fabric interconnect:

```
UCS-A# scope system
UCS-A/system # scope info-policy
UCS-A/system/info-policy # show
Info Policy:
State: Enabled
UCS-A/system/info-policy # disable
UCS-A/system/info-policy* # commit-buffer
UCS-A/system/info-policy #
```

Viewing the LAN Neighbors of the Fabric Interconnect

You must enable the information policy on the fabric interconnect to view the LAN neighbors.

SUMMARY STEPS

1. UCS-A# **scope fabric-interconnect {a | b}**
2. UCS-A/fabric-interconnect # **show lan-neighbors**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fabric-interconnect {a b}	Enters fabric interconnect mode for the specified fabric interconnect.
Step 2	UCS-A/fabric-interconnect # show lan-neighbors	Displays the fabric interconnect LAN neighbors.

Example

The following example shows how to display the LAN neighbors of the fabric interconnect:

```
UCS-A # scope fabric-interconnect a
UCS-A/fabric-interconnect # show lan-neighbors
Info Policy:Enabled
Lan Neighbors:
Local Interface: Ethernet1/2
Device Id: bgl-samc02-B (SSI140305YK)
IPv4 Address: 10.105.214.105
FI Port DN: sys/switch-A/slot-1/switch-ether/port-2
```

Viewing the SAN Neighbors of the Fabric Interconnect

You must enable the information policy on the fabric interconnect to view the SAN neighbors.

SUMMARY STEPS

1. UCS-A# **scope fabric-interconnect {a | b}**
2. UCS-A/fabric-interconnect # **show san-neighbors**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fabric-interconnect {a b}	Enters fabric interconnect mode for the specified fabric interconnect.
Step 2	UCS-A/fabric-interconnect # show san-neighbors	Displays the fabric interconnect SAN neighbors.

Example

The following example shows how to display the SAN neighbors of the fabric interconnect :

```
UCS-A # scope fabric-interconnect a
UCS-A/fabric-interconnect # show san-neighbors
Info Policy: Enabled
San neighbors:
Local Interface: fc2/1
Port VSAN: 100
Fabric Mgmt Addr: 10.65.124.252
Fabric pwnn: 20:02:00:05:9b:22:ad:C0
Fabric nwnn: 20:64:00:05:9b:22:ad:C1
My pwnn: 20:41:00:0d:ec:ee:dd:00
My nwnn: 20:64:00:0d:ec:ee:dd:01
FI Port DN: sys/switch-A/slot-2/switch-fc/port-1
```

Viewing the LLDP Neighbors of the Fabric Interconnect

You must enable the information policy on the fabric interconnect to view the LLDP neighbors.

SUMMARY STEPS

1. UCS-A# **scope fabric-interconnect {a | b}**
2. UCS-A/fabric-interconnect # **show lldp-neighbors**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fabric-interconnect {a b}	Enters fabric interconnect mode for the specified fabric interconnect.
Step 2	UCS-A/fabric-interconnect # show lldp-neighbors	Displays the fabric interconnect LLDP neighbors.

Example

The following example shows how to display the LLDP neighbors of the fabric interconnect :

```
UCS-A # scope fabric-interconnect a
UCS-A/fabric-interconnect # show lldp-neighbors
Info Policy: Enabled

Lldp Neighbors:

Local Interface: Eth1/5
Chassis Id: 000d.ecff.5e90
Remote Interface: Eth1/9
Remote Port Description: Ethernet1/9
System Name: bgl-samc02-B
System Description: Cisco Nexus Operating System (NX-OS) Software TAC support:
http://www.cisco.com/tac Copyright (c) 2002-2011, Cisco Systems, Inc
System Capabilities: B
Enabled Capabilities: B
Native VLAN: 1
IPv4 Mgmt Address: 10.105.214.105
FI Port DN: sys/switch-A/slot-1/switch-ether/port-5
```

Fabric Evacuation

Cisco UCS Manager introduces fabric evacuation, which is the ability to evacuate all traffic that flows through a fabric interconnect from all servers attached to it through an IOM or FEX while upgrading a system. Fabric evacuation is not supported on direct-attached rack servers.

Upgrading the secondary fabric interconnect in a system disrupts active traffic on the fabric interconnect. This traffic fails over to the primary fabric interconnect. You can use fabric evacuation during the upgrade process as follows:

1. Stop all the traffic that is active through a fabric interconnect.
2. For vNICs configured with failover, verify that the traffic has failed over by using Cisco UCS Manager, or tools such as vCenter.
3. Upgrade the secondary fabric interconnect.
4. Restart all the stopped traffic flows.
5. Change the cluster lead to the secondary fabric interconnect.
6. Repeat steps 1 to 4 and upgrade the primary fabric interconnect.

**Note**

- Fabric interconnect traffic evacuation is supported only in a cluster configuration.
- You can evacuate traffic only from the subordinate fabric interconnect.
- The IOM or FEX backplane ports of the fabric interconnect on which evacuation is configured will go down, and their state will appear as **Admin down**. During the manual upgrade process, to move these backplane ports back to the Up state and resume traffic flow, you must explicitly configure **Admin Evac Mode** as **Off**.
- Starting with Cisco UCS Manager Release 3.1(3), you can use fabric evacuation during Auto Install.
- If you use fabric evacuation outside of the upgrade process, you must re-acknowledge the FEX to get the VIFs back to the online state.

Stopping Traffic on a Fabric Interconnect

SUMMARY STEPS

1. UCS-A # **scope fabric-interconnect {a | b}**
2. UCS-A /fabric-interconnect # **stop server traffic [force]**
3. UCS-A /fabric-interconnect # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A # scope fabric-interconnect {a b}	Enters the fabric interconnect mode.
Step 2	UCS-A /fabric-interconnect # stop server traffic [force]	Stops all the traffic that is active through the specified Fabric Interconnect. Use the force option to evacuate a fabric interconnect regardless of its current evacuation state.
Step 3	UCS-A /fabric-interconnect # commit-buffer	Commits the transaction to the system configuration.

Example

This example shows how to stop all traffic that is active through Fabric Interconnect B:

```
UCS-A# scope fabric-interconnect b
UCS-A /fabric-interconnect # stop server traffic
Warning: Enabling fabric evacuation will stop all traffic through this Fabric Interconnect
        from servers attached through IOM/FEX. The traffic will fail over to the Primary Fabric
        Interconnect for fail over vnics.
UCS-A /fabric-interconnect # commit-buffer
```

Displaying the Status of Evacuation for a Fabric Interconnect

SUMMARY STEPS

1. UCS-A # **scope fabric-interconnect {a | b}**
2. UCS-A /fabric-interconnect # **show detail**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A # scope fabric-interconnect {a b}	Enters fabric interconnect mode for the specified fabric interconnect.
Step 2	UCS-A /fabric-interconnect # show detail	Displays details about the specified fabric interconnect.

Example

This example shows how to display the status of a fabric interconnect.



Note

Admin Evacuation and Oper Evacuation and show the status of evacuation at the fabric interconnect.

```
UCS-A /fabric-interconnect # show detail
```

```
Fabric Interconnect:
  ID: B
  Product Name: Cisco UCS 6248UP
  PID: UCS-FI-6248UP
  VID: V01
  Vendor: Cisco Systems, Inc.
  Serial (SN): SSI171400HG
  HW Revision: 0
  Total Memory (MB): 16165
  OOB IP Addr: 10.193.32.172
  OOB Gateway: 10.193.32.1
  OOB Netmask: 255.255.255.0
  OOB IPv6 Address: ::
  OOB IPv6 Gateway: ::
  Prefix: 64
  Operability: Operable
  Thermal Status: Ok
  Admin Evacuation: On
  Oper Evacuation: On
  Current Task 1:
  Current Task 2:
  Current Task 3:
```

Displaying the Status of Evacuation for an IOM

SUMMARY STEPS

1. UCS-A# **scope chassis** *chassis-num*
2. UCS-A /chassis # **scope iom** *iom-id*
3. UCS-A /chassis/iom # **show detail**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope chassis <i>chassis-num</i>	Enters chassis mode for the specified chassis.
Step 2	UCS-A /chassis # scope iom <i>iom-id</i>	Enters chassis IOM mode for the specified IOM.
Step 3	UCS-A /chassis/iom # show detail	Displays evacuation status details for the specified IOM.

Example

This example shows how to display the evacuation status details for an IOM.



Note

Oper Evacuation shows the operational status of evacuation for the IOM.

```
UCS-A# scope chassis 1
UCS-A /chassis # scope iom 1
UCS-A /chassis/iom # show detail

IOM:
  ID: 1
  Side: Left
  Fabric ID: A
  User Label:
  Overall Status: Fabric Conn Problem
  Oper qualifier: Server Port Problem
  Operability: Operable
  Presence: Equipped
  Thermal Status: OK
  Discovery: Online
  Config State: Ok
  Peer Comm Status: Connected
  Product Name: Cisco UCS 2204XP
  PID: UCS-IOM-2204XP
  VID: V02
  Part Number: 73-14488-02
  Vendor: Cisco Systems Inc
  Serial (SN): FCH1718J9FT
  HW Revision: 0
  Mfg Date: 2013-05-12T00:00:00.000
  Controller Subject: Iocard
  Fabric Port Aggregation Capability: Port Channel
  Oper Evacuation: On
  Current Task 1:
  Current Task 2:
```

Verifying Fabric Evacuation

SUMMARY STEPS

1. UCS-A# **show service-profile circuit server** *server-id*

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# show service-profile circuit server <i>server-id</i>	Shows the network circuit information for the service profile associated with the specified server.

Example

The following example shows the VIF (Virtual NIC) paths before fabric evacuation.



Note

- VIF at Fabric Interconnect A shows that traffic is initially active through the fabric interconnect.
- VIF at Fabric Interconnect B is passive before evacuation.

```
UCS-A# show service-profile circuit server 1/6
Service Profile: test1
Server: 1/6
  Fabric ID: A
    Path ID: 1
      VIF      vNIC      Link State  Oper State  Prot State  Prot Role  Admin
Pin  Oper Pin  Transport
-----
      692 eth0      Up          Active      Active      Primary    0/0
1/15  Ether
  Fabric ID: B
    Path ID: 1
      VIF      vNIC      Link State  Oper State  Prot State  Prot Role  Admin
Pin  Oper Pin  Transport
-----
      693 eth0      Up          Active      Passive     Backup     0/0
1/15  Ether
UCS-A#
```

The following example shows the VIF paths after Fabric Interconnect A is evacuated.

**Note**

- After failover, the VIF state at Fabric Interconnect A goes into error.
- VIF at Fabric Interconnect B takes over as active.

```
UCS-A# show service-profile circuit server 1/6
Service Profile: test1
Server: 1/6
  Fabric ID: A
    Path ID: 1
      VIF      vNIC      Link State  Oper State  Prot State    Prot Role  Admin
Pin  Oper Pin  Transport
-----
0/0      692 eth0      Error       Error       Active        Primary    0/0
      Ether
  Fabric ID: B
    Path ID: 1
      VIF      vNIC      Link State  Oper State  Prot State    Prot Role  Admin
Pin  Oper Pin  Transport
-----
1/15     693 eth0      Up          Active      Passive        Backup     0/0
      Ether
UCS-A#
```

Restarting Traffic on a Fabric Interconnect

SUMMARY STEPS

1. UCS-A # **scope fabric-interconnect {a | b}**
2. UCS-A /fabric-interconnect # **start server traffic**
3. UCS-A /fabric-interconnect # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A # scope fabric-interconnect {a b}	Enters the fabric interconnect mode.
Step 2	UCS-A /fabric-interconnect # start server traffic	Restarts traffic through the specified fabric interconnect.
Step 3	UCS-A /fabric-interconnect # commit-buffer	Commits the transaction to the system configuration.

Example

This example shows how to restart traffic through Fabric Interconnect B:

```
UCS-A# scope fabric-interconnect b
UCS-A /fabric-interconnect # start server traffic
Warning: Resetting fabric evacuation will cause server traffic that failed over to the
Primary Fabric Interconnect to fail back to this Fabric Interconnect.
```

```
UCS-A /fabric-interconnect # commit-buffer
```

Fabric Interconnect Port Types

By default, all fabric interconnect ports are unconfigured. For Ethernet LAN connectivity, fabric interconnect ports can be in the following states:

- **Unconfigured**—Port is not configured and cannot be used.
- **Server Port**—Port is configured for downlink connection to an IOM Fabric Extender (FEX) module in a blade chassis.
- **Uplink Port**—Port is configured for uplink connection to the upstream Ethernet switch. Uplink ports are always configured as trunk ports.
- **Disabled**—Port is configured either as an uplink or server port and is currently disabled by the administrator.

For 6200 series fabric interconnects, all ports are unified ports; therefore you also configure all the ports as 1/10 Gigabit Ethernet, Fibre Channel (FC), FC uplink, appliance port, or FCoE port.

For 6300 series fabric interconnects, see the *UCS Manager Getting Started Guide*.

For Cisco UCS 6400 Series Fabric Interconnects, ports 1 to 16 are unified ports and can be configured as either Ethernet or FC ports. *UCS Manager Getting Started Guide* has detailed information.



Note

The Cisco UCS 6454 Fabric Interconnect supported 8 unified ports (ports 1 - 8) with Cisco UCS Manager 4.0(1) and 4.0(2), but with release 4.0(4) and later it supports 16 unified ports (ports 1 - 16).

Fabric Interconnect Switching Modes

The Cisco UCS Fabric Interconnects operate in two main switching modes: Ethernet or Fibre Channel. These modes are independent of each other. They determine how the fabric interconnect behaves as a device between the server and network/server and storage device.

Ethernet Switching Mode

The Ethernet switching mode determines how the fabric interconnect behaves as a switching device between the servers and the network. The fabric interconnect operates in either of the following Ethernet switching modes:

End-Host Mode

End-host mode allows the fabric interconnect to act as an end host to the network, representing all servers (hosts) connected to it through vNICs. This behavior is achieved by pinning (either dynamically pinning or

hard pinning) vNICs to uplink ports, which provides redundancy to the network, and makes the uplink ports appear as server ports to the rest of the fabric.

In end-host mode, the fabric interconnect does not run the Spanning Tree Protocol (STP), but it avoids loops by denying uplink ports from forwarding traffic to each other and by denying egress server traffic on more than one uplink port at a time. End-host mode is the default Ethernet switching mode and should be used if either of the following is used upstream:

- Layer 2 switching for Layer 2 aggregation
- Virtual Switching System (VSS) aggregation layer



Note When you enable end-host mode, if a vNIC is hard pinned to an uplink port and this uplink port goes down, the system cannot repin the vNIC, and the vNIC remains down.

Switch Mode

Switch mode is the traditional Ethernet switching mode. The fabric interconnect runs STP to avoid loops, and broadcast and multicast packets are handled in the traditional way. Use the switch mode only if the fabric interconnect is directly connected to a router, or if either of the following is used upstream:

- Layer 3 aggregation
- VLAN in a box



Note For both Ethernet switching modes, even when vNICs are hard-pinned to uplink ports, all server-to-server unicast traffic in the server array is sent only through the fabric interconnect and is never sent through uplink ports. Server-to-server multicast and broadcast traffic is sent through all uplink ports in the same VLAN.

Cisco UCS Fabric Interconnect in Switch Mode with Cisco MDS 9000 Family Fibre Channel Switching Modules

While creating a port channel between a Cisco MDS 9000 family FC switching module and a Cisco UCS Fabric Interconnect in switch mode, use the following order:

1. Create the port channel on the MDS side.
2. Add the port channel member ports.
3. Create the port channel on the Fabric Interconnect side.
4. Add the port channel member ports.

If you create the port channel on the Fabric Interconnect side first, the ports will go into a suspended state.

When the Cisco UCS Fabric Interconnect is in switch mode, the port channel mode can only be in **ON** mode and not **Active**. However, to get the peer wwn information for the Fabric Interconnect, the port channel must be in **Active** mode.

Configuring Ethernet Switching Mode



Important

When you change the Ethernet switching mode, Cisco UCS Manager logs you out and restarts the fabric interconnect. For a cluster configuration, Cisco UCS Manager restarts both fabric interconnects. The subordinate fabric interconnect reboots first as a result of the change in switching mode. The primary fabric interconnect reboots only after you acknowledge it in **Pending Activities**. The primary fabric interconnect can take several minutes to complete the change in Ethernet switching mode and become system ready. The existing configuration is retained.

While the fabric interconnects are rebooting, all blade servers lose LAN and SAN connectivity, causing a complete outage of all services on the blades. This might cause the operating system to fail.

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **set mode {end-host | switch}**
3. UCS-A /eth-uplink # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # set mode {end-host switch}	Sets the fabric interconnect to the specified switching mode.
Step 3	UCS-A /eth-uplink # commit-buffer	Commits the transaction to the system configuration. Cisco UCS Manager restarts the fabric interconnect, logs you out, and disconnects Cisco UCS Manager CLI.

Example

The following example sets the fabric interconnect to end-host mode and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # set mode end-host
Warning: When committed, this change will cause the switch to reboot
UCS-A /eth-uplink* # commit-buffer
UCS-A /eth-uplink #
```

Fibre Channel Switching Mode

The Fibre Channel switching mode determines how the fabric interconnect behaves as a switching device between the servers and storage devices. The fabric interconnect operates in either of the following Fibre Channel switching modes:

End-Host Mode

End-host mode is synonymous with N Port Virtualization (NPV) mode. This mode is the default Fibre Channel Switching mode. End-host mode allows the fabric interconnect to act as an end host to the connected fibre channel networks, representing all servers (hosts) connected to it through virtual host bus adapters (vHBAs). This behavior is achieved by pinning (either dynamically pinning or hard-pinning) vHBAs to Fibre Channel uplink ports, which makes the Fibre Channel ports appear as server ports (N-ports) to the rest of the fabric. When in end-host mode, the fabric interconnect avoids loops by preventing uplink ports from receiving traffic from one another.



Note When you enable end-host mode, if a vHBA is hard-pinned to an uplink Fibre Channel port and this uplink port goes down, the system cannot repin the vHBA, and the vHBA remains down.

Switch Mode

Switch mode is not the default Fibre Channel switching mode. Switch mode allows the fabric interconnect to connect directly to a storage device. Enabling Fibre Channel switch mode is useful in Pod models where there is no SAN (for example, a single Cisco UCS domain that is connected directly to storage), or where a SAN exists (with an upstream MDS). In Fibre Channel switch mode, SAN pin groups are irrelevant. Any existing SAN pin groups are ignored.

Configuring Fibre Channel Switching Mode



Note When the Fibre Channel switching mode is changed, both Cisco UCS fabric interconnects reload simultaneously. Reloading the fabric interconnects will cause a system-wide downtime for approximately 10 to 15 minutes.

SUMMARY STEPS

1. UCS-A# **scope fc-uplink**
2. UCS-A /fc-uplink # **set mode {end-host | switch}**
3. UCS-A /fc-uplink # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fc-uplink	Enters Fibre Channel uplink mode.
Step 2	UCS-A /fc-uplink # set mode {end-host switch}	Sets the fabric interconnect to the specified switching mode.
Step 3	UCS-A /fc-uplink # commit-buffer	Commits the transaction to the system configuration. Cisco UCS Manager restarts the fabric interconnect, logs you out, and disconnects Cisco UCS Manager CLI.

Example

The following example shows how to set the fabric interconnect to end-host mode and commit the transaction:

```
UCS-A # scope fc-uplink
UCS-A /fc-uplink # set mode end-host
UCS-A /fc-uplink* # commit-buffer
UCS-A /fc-uplink #
```



CHAPTER 4

LAN Ports and Port Channels

- [Unified Ports on the Cisco UCS 6200 Series and 6324 Fabric Interconnects, on page 21](#)
- [Physical and Backplane Ports, on page 52](#)
- [Server Ports, on page 57](#)
- [Uplink Ethernet Ports, on page 59](#)
- [Appliance Ports, on page 62](#)
- [FCoE Uplink Ports, on page 69](#)
- [Unified Storage Ports, on page 73](#)
- [Unified Uplink Ports, on page 74](#)
- [FCoE and Fibre Channel Storage Ports, on page 75](#)
- [Uplink Ethernet Port Channels, on page 77](#)
- [Appliance Port Channels, on page 81](#)
- [Fibre Channel Port Channels, on page 86](#)
- [FCoE Port Channels, on page 92](#)
- [Unified Uplink Port Channel, on page 94](#)
- [Event Detection and Action, on page 95](#)
- [Adapter Port Channels, on page 100](#)
- [Fabric Port Channels, on page 101](#)

Unified Ports on the Cisco UCS 6200 Series and 6324 Fabric Interconnects

Unified ports are ports on the Cisco UCS 6200 Series and 6324 Fabric Interconnects that you can configure to carry either Ethernet or Fibre Channel traffic. A Cisco UCS domain cannot use these un-reserved ports until you configure them.



Note

When you configure a port on a fabric interconnect, the administrative state is automatically set to enabled. If the port is connected to another device, this may cause traffic disruption. You can disable the port after configuring it. Configurable beacon LEDs indicate which unified ports are configured for the selected port mode.

Port Modes

The port mode determines whether a unified port on the fabric interconnect is configured to carry Ethernet or Fibre Channel traffic. You configure the port mode in Cisco UCS Manager. However, the fabric interconnect does not automatically discover the port mode.

Changing the port mode deletes the existing port configuration and replaces it with a new logical port. Any objects associated with that port configuration, such as VLANs and VSANS, are also removed. There is no restriction on the number of times you can change the port mode for a unified port.

Port Types

The port type defines the type of traffic carried over a unified port connection.

By default, unified ports changed to Ethernet port mode are set to the Ethernet uplink port type. Unified ports changed to Fibre Channel port mode are set to the Fibre Channel uplink port type. You cannot unconfigure Fibre Channel ports.

Changing the port type does not require a reboot.

Ethernet Port Mode

When you set the port mode to Ethernet, you can configure the following port types:

- Server ports
- Ethernet uplink ports
- Ethernet port channel members
- FCoE ports
- Appliance ports
- Appliance port channel members
- SPAN destination ports
- SPAN source ports

**Note**

For SPAN source ports, configure one of the port types and then configure the port as SPAN source.

Fibre Channel Port Mode

When you set the port mode to Fibre Channel, you can configure the following port types:

- Fibre Channel uplink ports
- Fibre Channel port channel members
- Fibre Channel storage ports
- FCoE Uplink ports
- SPAN source ports



Note For SPAN source ports, configure one of the port types and then configure the port as SPAN source.

Data Traffic Interruption from Port Mode Changing

Port mode changes can cause an interruption to the data traffic for the Cisco UCS domain. The length of the interruption and the affected traffic depend upon the configuration of the Cisco UCS domain and the module on which you made the port mode changes.



Tip To minimize traffic disruption during system changes, form a Fibre Channel uplink port-channel across the fixed and expansion modules.

Impact of Port Mode on an Expansion Module

After you make port mode changes on an expansion module, the module reboots. All traffic through port on the expansion module is interrupted for approximately 1 minute while the module reboots.

Impact of Port Mode Changes on the Fixed Module in a Cluster Configuration

A cluster configuration has two fabric interconnects. After you make port changes to the fixed module, the fabric interconnect reboots. The impact on the data traffic depends upon whether or not you have configured the server vNICs to failover to the other fabric interconnect when one fails.

If you change the port modes on the expansion module of one fabric interconnect and then wait for that to reboot before changing the port modes on the second fabric interconnect, the following occurs:

- With server vNIC failover, traffic fails over to the other fabric interconnect and no interruption occurs.
- Without server vNIC failover, all data traffic through the fabric interconnect on which you changed the port modes is interrupted for approximately eight minutes while the fabric interconnect reboots.

When you change the port modes on the fixed modules of both fabric interconnects simultaneously, all data traffic through the fabric interconnects are interrupted for approximately eight minutes while the fabric interconnects reboot.

Impact of Port Mode Changes on the Fixed Module in a Standalone Configuration

A standalone configuration has only one fabric interconnect. After you make port changes to the fixed module, the fabric interconnect reboots. All data traffic through the fabric interconnect is interrupted for approximately eight minutes while the fabric interconnect reboots.

Guidelines for Configuring Unified Ports

Consider the following guidelines and restrictions when configuring unified ports:

Hardware and Software Requirements

Unified ports are supported on the 6200 series fabric interconnect with Cisco UCS Manager, version 2.0.

Unified ports are not supported on 6100 series fabric interconnects, even if they are running Cisco UCS Manager, version 2.0.

Port Mode Placement

Because the Cisco UCS Manager GUI interface uses a slider to configure the port mode for unified ports on a fixed or expansion module, it automatically enforces the following restrictions which limits how port modes can be assigned to unified ports. When using the Cisco UCS Manager CLI interface, these restrictions are enforced when you commit the transaction to the system configuration. If the port mode configuration violates any of the following restrictions, the Cisco UCS Manager CLI displays an error:

- Ethernet ports must be grouped together in a block. For each module (fixed or expansion), the Ethernet port block must start with the first port and end with an even numbered port.
- Fibre Channel ports must be grouped together in a block. For each module (fixed or expansion), the first port in the Fibre Channel port block must follow the last Ethernet port and extend to include the rest of the ports in the module. For configurations that include only Fibre Channel ports, the Fibre Channel block must start with the first port on the fixed or expansion module.
- Alternating Ethernet and Fibre Channel ports is not supported on a single module.

Example of a valid configuration— Might include unified ports 1–16 on the fixed module configured in Ethernet port mode and ports 17–32 in Fibre Channel port mode. On the expansion module you could configure ports 1–4 in Ethernet port mode and then configure ports 5–16 in Fibre Channel mode. The rule about alternating Ethernet and Fibre Channel port types is not violated because this port arrangement complies with the rules on each individual module.

Example of an invalid configuration— Might include a block of Fibre Channel ports starting with port 16. Because each block of ports has to start with an odd-numbered port, you would have to start the block with port 17.

The total number of uplink Ethernet ports and uplink Ethernet port channel members that can be configured on each fabric interconnect is limited to 31. This limitation includes uplink Ethernet ports and uplink Ethernet port channel members configured on the expansion module.

Special Considerations for UCS Manager CLI Users

Because the Cisco UCS Manager CLI does not validate port mode changes until you commit the buffer to the system configuration, it is easy to violate the grouping restrictions if you attempt to commit the buffer before creating at least two new interfaces. To prevent errors, we recommend that you wait to commit your changes to the system configuration until you have created new interfaces for all of the unified ports changing from one port mode to another.

Committing the buffer before configuring multiple interfaces will result in an error, but you do not need to start over. You can continue to configure unified ports until the configuration satisfies the aforementioned requirements.

Cautions and Guidelines for Configuring Unified Uplink Ports and Unified Storage Ports

The following are cautions and guidelines to follow while working with unified uplink ports and unified storage ports:

- In an unified uplink port, if you enable one component as a SPAN source, the other component will automatically become a SPAN source.

**Note**

If you create or delete a SPAN source under the Ethernet uplink port, Cisco UCS Manager automatically creates or deletes a SPAN source under the FCoE uplink port. The same happens when you create a SPAN source on the FCoE uplink port.

- You must configure a non default native VLAN on FCoE and unified uplink ports. This VLAN is not used for any traffic. Cisco UCS Manager will reuse an existing fcoe-storage-native-vlan for this purpose. This fcoe-storage-native-vlan will be used as a native VLAN on FCoE and unified uplinks.
- In an unified uplink port, if you do not specify a non default VLAN for the Ethernet uplink port the fcoe-storage-native-vlan will be assigned as the native VLAN on the unified uplink port. If the Ethernet port has a non default native VLAN specified as native VLAN, this will be assigned as the native VLAN for unified uplink port.
- When you create or delete a member port under an Ethernet port channel, Cisco UCS Manager automatically creates or deletes the member port under FCoE port channel. The same happens when you create or delete a member port in FCoE port channel.
- When you configure an Ethernet port as a standalone port, such as server port, Ethernet uplink, FCoE uplink or FCoE storage and make it a member port for an Ethernet or FCoE port channel, Cisco UCS Manager automatically makes this port a member of both Ethernet and FCoE port channels.
- When you remove the membership for a member port from being a member of server uplink, Ethernet uplink, FCoE uplink or FCoE storage, Cisco UCS Manager deletes the corresponding members ports from Ethernet port channel and FCoE port channel and creates a new standalone port.
- If you downgrade Cisco UCS Manager from release 2.1 to any of the prior releases, all unified uplink ports and port channels will be converted to Ethernet ports and Ethernet port channels when the downgrade is complete. Similarly, all the unified storage ports will be converted to appliance ports.
- For unified uplink ports and unified storage ports, when you create two interfaces, only one license is checked out. As long as either interface is enabled, the license remains checked out. The license will be released only if both the interfaces are disabled for a unified uplink port or a unified storage port.
- Cisco UCS 6100 series fabric interconnect switch can only support 1VF or 1VF-PO facing same downstream NPV switch.

Configuring the Port Mode



Caution

Changing the port mode on either module can cause an interruption in data traffic because changes to the fixed module require a reboot of the fabric interconnect and changes on an expansion module require a reboot of that module.

If the Cisco UCS domain has a cluster configuration that is set up for high availability and servers with service profiles that are configured for failover, traffic fails over to the other fabric interconnect and data traffic is not interrupted when the port mode is changed on the fixed module.

In the Cisco UCS Manager CLI, there are no new commands to support Unified Ports. Instead, you change the port mode by scoping to the mode for the desired port type and then creating a new interface. When you create a new interface for an already configured slot ID and port ID, UCS Manager deletes the previously configured interface and creates a new one. If a port mode change is required because you configure a port that previously operated in Ethernet port mode to a port type in Fibre Channel port mode, UCS Manager notes the change.

Expansions modules are not supported with Cisco UCS Mini.

SUMMARY STEPS

1. UCS-A# **scope** *port-type-mode*
2. UCS-A /*port-type-mode* # **scope fabric** {a | b}
3. UCS-A /*port-type-mode*/fabric # **create interface** *slot-id port-id*
4. Create new interfaces for other ports belonging to the Ethernet or Fibre Channel port block.
5. UCS-A /*port-type-mode*/fabric/interface # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope <i>port-type-mode</i>	<p>Enters the specified port type mode for one of the following port types:</p> <p>eth-server</p> <p>For configuring server ports.</p> <p>eth-storage</p> <p>For configuring Ethernet storage ports and Ethernet storage port channels.</p> <p>eth-traffic-mon</p> <p>For configuring Ethernet SPAN ports.</p> <p>eth-uplink</p> <p>For configuring Ethernet uplink ports.</p> <p>fc-storage</p> <p>For configuring Fibre Channel storage ports.</p>

	Command or Action	Purpose
		fc-traffic-mon For configuring Fibre Channel SPAN ports. fc-uplink For configuring Fibre Channel uplink ports and Fibre Channel uplink port channels.
Step 2	UCS-A <i>/port-type-mode</i> # scope fabric {a b}	Enters the specified port type mode for the specified fabric.
Step 3	UCS-A <i>/port-type-mode/fabric</i> # create interface slot-id port-id	Creates an interface for the specified port type. If you are changing the port type from Ethernet port mode to Fibre Channel port mode, or vice-versa, the following warning appears: Warning: This operation will change the port mode (from Ethernet to FC or vice-versa). When committed, this change will require the module to restart.
Step 4	Create new interfaces for other ports belonging to the Ethernet or Fibre Channel port block.	There are several restrictions that govern how Ethernet and Fibre Channel ports can be arranged on a fixed or expansion module. Among other restrictions, it is required that you change ports in groups of two. Violating any of the restrictions outlined in the <i>Guidelines and Recommendations for Configuring Unified Ports</i> section will result in an error.
Step 5	UCS-A <i>/port-type-mode/fabric/interface</i> # commit-buffer	Commits the transaction to the system configuration.

Based on the module for which you configured the port modes, data traffic for the Cisco UCS domain is interrupted as follows:

- Fixed module—The fabric interconnect reboots. All data traffic through that fabric interconnect is interrupted. In a cluster configuration that provides high availability and includes servers with vNICs that are configured for failover, traffic fails over to the other fabric interconnect and no interruption occurs. Changing the port mode for both sides at once results in both fabric interconnects rebooting simultaneously and a complete loss of traffic until both fabric interconnects are brought back up.

It takes about 8 minutes for the fixed module to reboot.

- Expansion module—The module reboots. All data traffic through ports in that module is interrupted.

It takes about 1 minute for the expansion module to reboot.

Example

The following example changes ports 3 and 4 on slot 1 from Ethernet uplink ports in Ethernet port mode to uplink Fibre Channel ports in Fibre Channel port mode:

```
UCS-A# scope fc-uplink
UCS-A /fc-uplink # scope fabric a
UCS-A /fc-uplink/fabric # create interface 1 3
Warning: This operation will change the port mode (from Ethernet to FC or vice-versa).
```

```

When committed, this change will require the fixed module to restart.
UCS-A /fc-uplink/fabric/interface* # up
UCS-A /fc-uplink/fabric* #create interface 1 4
Warning: This operation will change the port mode (from Ethernet to FC or vice-versa).
When committed, this change will require the fixed module to restart.
UCS-A /fc-uplink/fabric/interface* #commit-buffer

```

Configuring Breakout Ports

Port Breakout Functionality on Cisco UCS 6454 Fabric Interconnects

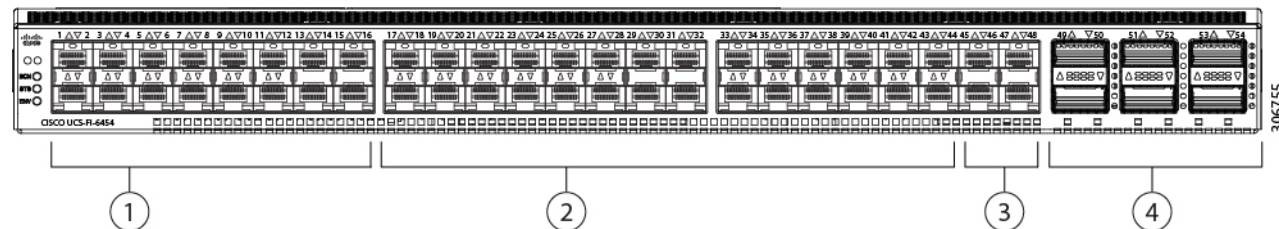
About Breakout Ports

Cisco UCS 6454 fabric interconnects support splitting a single 40/100G QSFP port into four 10/25G ports using a supported breakout cable. These ports can be used only as uplink ports connecting to a 10/25G switch. On the UCS 6454 fabric interconnect, by default, there are 6 ports in the 40/100G mode. These are ports 49 to 54. These 40/100G ports are numbered in a 2-tuple naming convention. For example, the second 40G port is numbered as 1/50. The process of changing the configuration from 40G to 10 G, or from 100G to 25G is called breakout, and the process of changing the configuration from [4X]10G to 40G or from [4X]25G to 100G is called unconfigure.

When you break out a 40G port into 10G ports or a 100G port into 25G ports, the resulting ports are numbered using a 3-tuple naming convention. For example, the breakout ports of the second 40-Gigabit Ethernet port are numbered as 1/50/1, 1/50/2, 1/50/3, 1/50/4.

The following image shows the rear view of the Cisco UCS 6454 fabric interconnect, and includes the ports that support breakout port functionality:

Figure 1: Cisco UCS 6454 Fabric Interconnect Rear View



1	Ports 1-16 (Unified Ports 10/25 Gbps Ethernet or FCoE or 8/16/32 Gbps Fibre Channel)	2	Ports 17-44 (10/25 Gbps Ethernet or FCoE)
3	Ports 45-48 (1/10/25 Gbps Ethernet or FCoE)	4	Uplink Ports 49-54 (40/100 Gbps Ethernet or FCoE)

Breakout Port Guidelines

The following are the guidelines for breakout functionality for Cisco UCS 6454 fabric interconnects:

- The breakout configurable ports are ports 49-54.
- You cannot configure the speed for each breakout port. Each breakout port is in auto mode.

- The fabric interconnect is rebooted after you configure the breakout mode for any of the supported fabric interconnect ports (1/49 to 1/54).
- In Cisco UCS Manager Release 4.0(2), breakout ports are not supported as destinations for traffic monitoring.
- Ports 49-54 can only be configured as uplink ports. They cannot be configured as any of the following:
 - Server ports
 - FCoE storage ports
 - Appliance ports

Port Breakout Functionality on Cisco UCS 64108 Fabric Interconnects

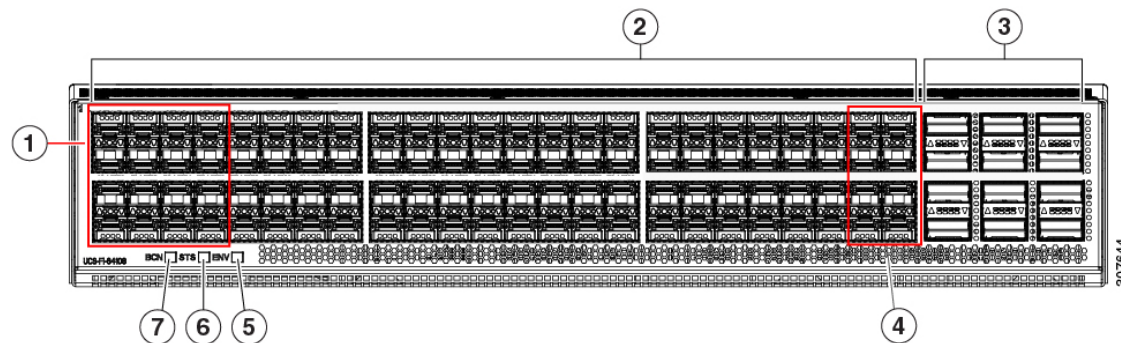
About Breakout Ports

Cisco UCS 64108 fabric interconnects support splitting a single 40/100G QSFP port into four 10/25G ports using a supported breakout cable. On the UCS 64108 fabric interconnect, by default, there are 12 ports in the 40/100G mode. These are ports 97 to 108. These 40/100G ports are numbered in a 2-tuple naming convention. For example, the second 40G port is numbered as 1/99. The process of changing the configuration from 40G to 10 G, or from 100G to 25G is called breakout, and the process of changing the configuration from [4X]10G to 40G or from [4X]25G to 100G is called unconfigure. These ports can be used as uplink, appliance, and FCoE storage ports. They cannot be configured as server ports.

When you break out a 40G port into 10G ports or a 100G port into 25G ports, the resulting ports are numbered using a 3-tuple naming convention. For example, the breakout ports of the second 40-Gigabit Ethernet port are numbered as 1/99/1, 1/99/2, 1/99/3, 1/99/4.

The following image shows the rear view of the Cisco UCS 64108 fabric interconnect, and includes the ports that support breakout port functionality:

Figure 2: Cisco UCS 64108 Fabric Interconnect Rear View



1	Ports 1-16. Unified Ports can operate as 10/25 Gbps Ethernet or 8/16/32 Gbps Fibre Channel. FC ports are converted in groups of four. Unified ports: <ul style="list-style-type: none"> • 10/25 Gbps Ethernet or FCoE • 8/16/32 Gbps Fibre Channel 	2	Ports 1-96. Each port can operate as either a 10 Gbps or 25 Gbps Ethernet or FCoE SFP28 port.
3	Uplink Ports 97-108. Each port can operate as either a 40 Gbps or 100 Gbps Ethernet or FCoE port. When using a breakout cable, each of these ports can operate as 4 x 10 Gbps or 4 x 25 Gbps Ethernet or FCoE ports. Ports 97 - 108 can be used to connect to Ethernet or FCoE uplink ports, and not to UCS server ports.	4	Ports 89-96 <ul style="list-style-type: none"> • 10/25 Gbps Ethernet or FCoE • 1 Gbps Ethernet
5	System environment (fan fault) LED	6	System status LED
7	Beacon LED		

Breakout Port Guidelines

The following are the guidelines for breakout functionality for Cisco UCS 64108 fabric interconnects:

- The breakout configurable ports are ports 97-108.
- You cannot configure the speed for each breakout port. Each breakout port is in auto mode.
- The fabric interconnect is rebooted after you configure the breakout mode for any of the supported fabric interconnect ports (1/97 to 1/108).
- Breakout ports are not supported as destinations for traffic monitoring.
- Ports 97-108 can be configured as uplink, appliance, and FCoE storage ports. They cannot be configured as server ports.

Port Breakout Functionality on Cisco UCS 6300 Series Fabric Interconnects

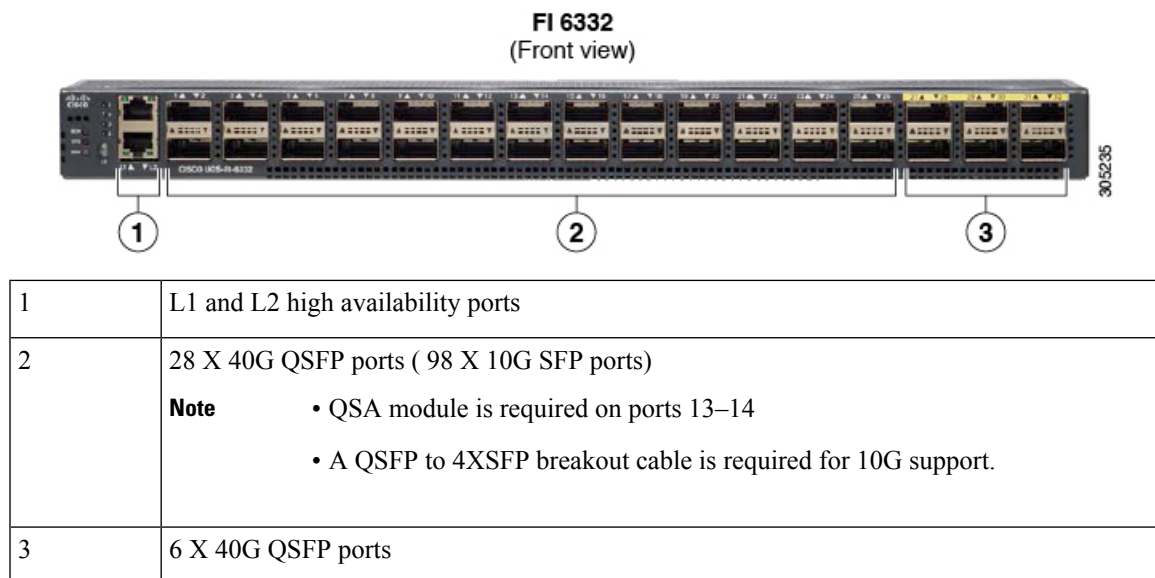
About Breakout Ports

Cisco UCS fabric interconnect 6300 series supports splitting a single QSFP port into four 10G ports using a supported breakout cable. By default, there are 32 ports in the 40G mode. These 40G ports are numbered in a 2-tuple naming convention. For example, the second 40G port is numbered as 1/2. The process of changing the configuration from 40G to 10G is called breakout and the process of changing the configuration from [4X]10G to 40G is called unconfigure.

When you break out a 40G port into 10G ports, the resulting ports are numbered using a 3-tuple naming convention. For example, the breakout ports of the second 40-Gigabit Ethernet port are numbered as 1/2/1, 1/2/2, 1/2/3, 1/2/4.

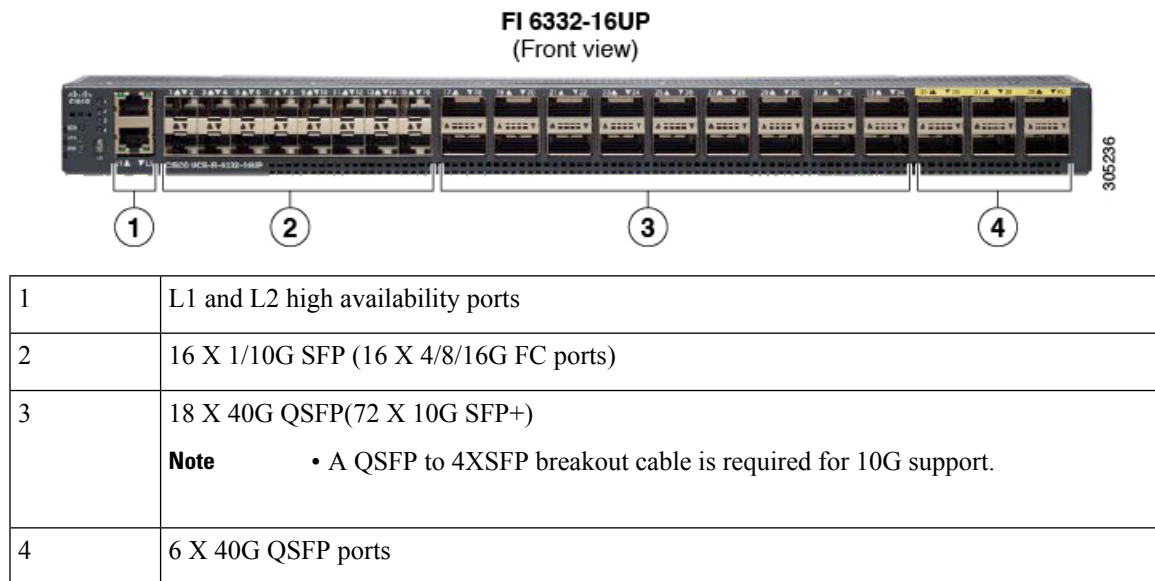
The following image shows the front view for the Cisco UCS 6332 series fabric interconnects, and includes the ports that may support breakout port functionality:

Figure 3: Cisco UCS 6332 Series Fabric Interconnects Front View



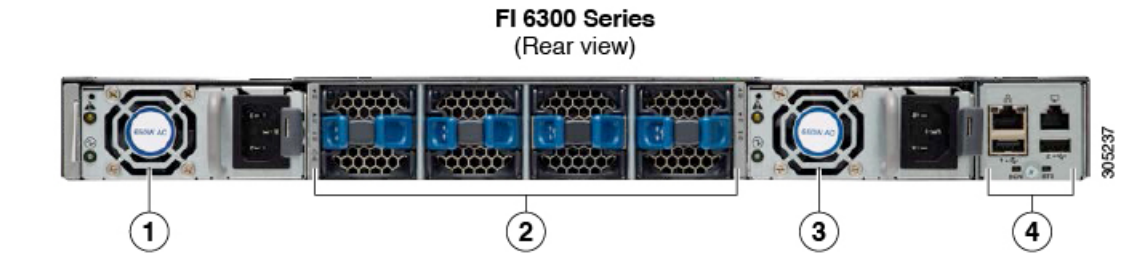
The following image shows the front view for the Cisco UCS 6332-16UP series fabric interconnects, and includes the ports that may support breakout port functionality:

Figure 4: Cisco UCS 6332-16UP Series Fabric Interconnects Front View



The following image shows the rear view of the Cisco UCS 6300 series fabric interconnects.

Figure 5: Cisco UCS 6300 Series Fabric Interconnects Rear View



1	Power supply
2	Four fans
3	Power supply
4	Serial ports

Breakout Port Constraints

The following table summarizes the constraints for breakout functionality for Cisco UCS 6300 series fabric interconnects:

Cisco UCS 6300 Series Fabric Interconnect Series	Breakout Configurable Ports	Ports without breakout functionality support
Cisco UCS 6332	1–12, 15–26	13–14, 27–32 Note • Auto-negotiate behavior is not supported on ports 27–32.
Cisco UCS 6332-16UP	17–34	1–16, 35–40 Note • Auto-negotiate behavior is not supported on ports 35–40



Important

Up to four breakout ports are allowed if QoS jumbo frames are used.

Configuring Multiple Breakout Ports

On a UCS 6300 Fabric Interconnect, you can specify a 40 Gigabit Ethernet port and create four 10 Gigabit Ethernet unconfigured breakout ports. On a UCS 6454 Fabric Interconnect, you can specify a 40 or 100 Gigabit Ethernet port and create four 10 or 25 Gigabit Ethernet unconfigured breakout ports. Because configuring breakout on a port causes the reboot of the Fabric Interconnect, we recommend that you breakout all required ports in a single transaction.

Before you begin

Before configuring a breakout port, view the port status using the **show port** command.

SUMMARY STEPS

1. UCS-A # **scope cabling**
2. UCS-A /cabling # **scope fabric {a | b}**
3. UCS-A /cabling/fabric # **create breakout slot-id port-id**
4. UCS-A /cabling/fabric/breakout* # **set breakouttype {10g-4x | 25g-4x}**
5. UCS-A /cabling/fabric/breakout* # **up**
6. UCS-A /cabling/fabric/breakout* # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A # scope cabling	Enters the cabling mode.
Step 2	UCS-A /cabling # scope fabric {a b}	Enters cabling fabric mode for the specified fabric.
Step 3	UCS-A /cabling/fabric # create breakout slot-id port-id	Creates the breakout port on the selected slot and port.
Step 4	UCS-A /cabling/fabric/breakout* # set breakouttype {10g-4x 25g-4x}	Specifies the type of breakout port on a UCS 6454 Fabric Interconnect.
Step 5	UCS-A /cabling/fabric/breakout* # up	Returns you to fabric mode. Repeat steps 3 and 5 for each breakout port on a UCS 6300, and repeat steps 3, 4 and 5 for each breakout port on a UCS 6454.
Step 6	UCS-A /cabling/fabric/breakout* # commit-buffer	Commits the transaction to the server.

Example

The following example creates breakout ports 1/1 through 1/4 on a UCS 6300 Fabric Interconnect and commits the transaction:

```
UCS-A# scope cabling
UCS-A /cabling # scope fabric a
UCS-A /cabling/fabric # create breakout 1 1
Warning: Port breakout create action reboots FI and any existing configurations on 40G port
will be erased.
UCS-A /cabling/fabric/breakout* # up
UCS-A /cabling/fabric* # create breakout 1 2
Warning: Port breakout create action reboots FI and any existing configurations on 40G port
will be erased.
UCS-A /cabling/fabric/breakout* # up
UCS-A /cabling/fabric* # create breakout 1 3
Warning: Port breakout create action reboots FI and any existing configurations on 40G port
will be erased.
UCSM--A /cabling/fabric/breakout* # up
UCSM-shiva-a-A /cabling/fabric* # create breakout 1 4
Warning: Port breakout create action reboots FI and any existing configurations on 40G port
will be erased.!
```

Configuring a Breakout Ethernet Uplink Port

```
UCSM--A /cabling/fabric/breakout* # commit-buffer
```

The following example creates breakout ports 1/49 through 1/52 on a UCS 6454 Fabric Interconnect, sets the breakout type, and commits the transaction:

```
UCS-A# scope cabling
UCS-A /cabling # scope fabric a
UCS-A /cabling/fabric # create breakout 1 49
Warning: Port breakout create action reboots FI and any existing configurations on 40G port
will be erased.!
UCS-A /cabling/fabric/breakout* # set breakouttype 10g-4x
UCS-A /cabling/fabric/breakout* # up
UCS-A /cabling/fabric* # create breakout 1 50
Warning: Port breakout create action reboots FI and any existing configurations on 40G port
will be erased.!
UCS-A /cabling/fabric/breakout* # set breakouttype 10g-4x
UCS-A /cabling/fabric/breakout* # up
UCS-A /cabling/fabric* # create breakout 1 51
Warning: Port breakout create action reboots FI and any existing configurations on 40G port
will be erased.!
UCS-A /cabling/fabric/breakout* # set breakouttype 10g-4x
UCSM--A /cabling/fabric/breakout* # up
UCSM-shiva-a-A /cabling/fabric* # create breakout 1 52
Warning: Port breakout create action reboots FI and any existing configurations on 40G port
will be erased.!
UCS-A /cabling/fabric/breakout* # set breakouttype 10g-4x
UCS-A /cabling/fabric/breakout* # commit-buffer
```

What to do next

Verify that you created breakout ports on the fabric interconnect and on the NXOS switch. On the fabric interconnect use the **show breakout** command in cabling fabric mode for the specified fabric. In NXOS, use the **show interface brief** command.

Configuring a Breakout Ethernet Uplink Port

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **scope fabric{a | b}**
3. UCS-A /eth-uplink/fabric # **create aggr-interface slot-numaggregate port-num**
4. UCS-A /eth-uplink/fabric/aggr-interface* # **create br-interface breakout-port-num**
5. UCS-A /eth-uplink/fabric/aggr-interface/br-interface # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # scope fabric{a b}	Enters Ethernet uplink fabric mode for the specified fabric.
Step 3	UCS-A /eth-uplink/fabric # create aggr-interface slot-numaggregate port-num	Creates the interface for the specified aggregate (main) Ethernet uplink port.

	Command or Action	Purpose
Step 4	UCS-A /eth-uplink/fabric/aggr-interface* # create br-interface breakout-port-num	Creates an interface for the specified breakout Ethernet uplink port.
Step 5	UCS-A /eth-uplink/fabric/aggr-interface/br-interface # commit-buffer	Commits the transaction to the server.

Example

The following example shows how to create an interface for breakout Ethernet uplink port 1 of the aggregate port 21 on slot 1 of fabric A:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope fabric a
UCS-A /eth-uplink/fabric # enter aggr-interface 1 21
UCS-A /eth-uplink/fabric/aggr-interface # create br-interface 1
UCS-A /eth-uplink/fabric/aggr-interface/br-interface*# commit-buffer
```

The following example shows how to create interfaces for breakout Ethernet uplink ports 1-4 of the aggregate port 49 on slot 1 of fabric A on a UCS 6454 fabric interconnect, and commit the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope fabric a
UCS-A /eth-uplink/fabric # create aggr-interface 1 49
UCS-A /eth-uplink/fabric/aggr-interface* # create br-interface 1
UCS-A /eth-uplink/fabric/aggr-interface/br-interface* # up
UCS-A /eth-uplink/fabric/aggr-interface* # create br-interface 2
UCS-A /eth-uplink/fabric/aggr-interface/br-interface* # up
UCS-A /eth-uplink/fabric/aggr-interface* # create br-interface 3
UCS-A /eth-uplink/fabric/aggr-interface/br-interface* # up
UCS-A /eth-uplink/fabric/aggr-interface* # create br-interface 4
UCS-A /eth-uplink/fabric/aggr-interface/br-interface* # up
UCS-A /eth-uplink/fabric/aggr-interface* # commit-buffer
UCS-A /eth-uplink/fabric/aggr-interface #
```

The following example shows the breakout configuration for ports 1/49/1 to 1/49/4 of fabric A on a UCS 6454 fabric interconnect:

```
UCS-A# scope fabric-interconnect a
UCS-A /fabric-interconnect # show port
Ether Port:
```

Slot	Aggr	Port	Port Oper	State	Mac	Role	Xcvr
1	49	1	Sfp Not	Present	8C:60:4F:BC:C4:D4	Unknown	N/A
1	49	2	Sfp Not	Present	8C:60:4F:BC:C4:D5	Unknown	N/A
1	49	3	Sfp Not	Present	8C:60:4F:BC:C4:D6	Unknown	N/A
1	49	4	Sfp Not	Present	8C:60:4F:BC:C4:D7	Unknown	N/A

Configuring a Breakout Ethernet Uplink Port Channel Member

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A# /eth-uplink # **scope fabric {a | b}**
3. UCS-A# /eth-uplink/fabric # **scope fcoe-port-channel fcoe-port-channel**

4. UCS-A /eth-uplink/fabric/port-channel/fcoe-port-channel # **enter aggr-interface** *slot-id port-id*
5. UCS-A /eth-uplink/fabric/port-channel/member-aggr-port # **create br-member-port***breakout-port-num*
6. UCS-A /eth-uplink/fabric/port-channel/member-aggr-port/br-member-port # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A# /eth-uplink # scope fabric {a b}	Enters Ethernet uplink mode for the specified fabric.
Step 3	UCS-A# /eth-uplink/fabric # scope fcoe-port-channel <i>fcoe-port-channel</i>	Enters port channel for the specified FCoE uplink port.
Step 4	UCS-A /eth-uplink/fabric/port-channel/fcoe-port-channel # enter aggr-interface <i>slot-id port-id</i>	Enters the interface for the specified aggregate(main) FCoE uplink port.
Step 5	UCS-A /eth-uplink/fabric/port-channel/member-aggr-port # create br-member-port <i>breakout-port-num</i>	Creates the FCoE uplink port channel member.
Step 6	UCS-A /eth-uplink/fabric/port-channel/member-aggr-port/br-member-port # commit-buffer Example: The following example creates an Ethernet uplink port channel member for an Ethernet port on port 2, and commits the transaction: UCS-A# scope eth-storage UCS-A /eth-uplink # scope fabric a UCS-A /eth-uplink/fabric # scope fcoe-port-channel 51 UCS-A /eth-uplink/fabric/port-channel/member-aggr-port # create br-member-port 2 UCS-A /eth-uplink/fabric/port-channel/member-aggr-port/br-member-port* # commit-buffer	Commits the transaction to the server.

Configuring Ethernet Uplink Breakout Port as a Pin Group Target

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A# /eth-uplink/pin-group # **enter pin-group** *pin-group-name*
3. UCS-A# /eth-uplink/pin-group # **set target** {a|b}
breakout-ports*slot-numaggregate-port-numbreakout-port-num*
4. UCS-A # /eth-uplink/pin-group # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A# /eth-uplink/pin-group # enter pin-group <i>pin-group-name</i>	Enters the pin group with the specified name.
Step 3	UCS-A# /eth-uplink/pin-group # set target {a b} breakout-ports <i>slot-numaggregate-port-numbreakout-port-num</i>	Sets the selected target as the breakout port.
Step 4	UCS-A # /eth-uplink/pin-group # commit-buffer Example: The following example sets the pin group target to breakout port 2 of the aggregate port 1 on slot 1, on fabric A , and commits the transaction: UCS-A# scope eth-uplink UCS-A /eth-uplink # enter pin-group test UCS-A /eth-uplink/pin-group # set target a breakout-port 1 1 2 UCS-A /eth-uplink/pin-group* # commit-buffer	Commits the transaction to the server.

Configuring Breakout Appliance Ports

SUMMARY STEPS

1. UCS-A# **scope eth-storage**
2. UCS-A# /eth-storage # **scope fabric {a | b}**
3. UCS-A# /eth-storage/fabric # **enter aggr-interface** *slot-numaggregate-port-num*
4. UCS-A# /eth-storage/fabric/port-channel/member-aggr-port # **create br -interface***breakout-port-num*
5. UCS-A# /eth-storage/fabric/port-channel/member-aggr-port/br-member-port # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-storage	Enters Ethernet storage mode.
Step 2	UCS-A# /eth-storage # scope fabric {a b}	Enters Ethernet storage mode for the specified fabric.
Step 3	UCS-A# /eth-storage/fabric # enter aggr-interface <i>slot-numaggregate-port-num</i>	Enters the interface for the specified aggregate(main) appliance port.
Step 4	UCS-A# /eth-storage/fabric/port-channel/member-aggr-port # create br -interface <i>breakout-port-num</i>	Creates an interface for the specified breakout appliance port.
Step 5	UCS-A# /eth-storage/fabric/port-channel/member-aggr-port/br-member-port # commit-buffer	Commits the transaction to the server.

	Command or Action	Purpose
	<p>Example:</p> <p>The following example creates an interface for an appliance port 1 of the aggregate port 20 on slot 1 of fabric B, and commits the transaction:</p> <pre> UCS-A# scope eth-storage UCS-A /eth-storage # scope fabric a UCS-A /eth-storage/fabric # enter aggr-interface 1 20 UCS-A /eth-storage/fabric/aggr-interface # create br-interface 1 UCS-A /eth-storage/fabric/aggr-interface/br-interface* # commit-buffer </pre>	

Configuring a Breakout Appliance Port Channel Member

SUMMARY STEPS

1. UCS-A# **scope eth-storage**
2. UCS-A# /eth-storage # **scope fabric {a | b}**
3. UCS-A# /eth-storage # **scope port-channel***port-channel-num*
4. UCS-A# /eth-storage/fabric # **enter aggr-interface** *slot-numaggregate-port-num*
5. UCS-A /eth-storage/fabric/port-channel # **enter member-aggr-port** *slot-id port-id*
6. UCS-A /eth-storage/fabric/port-channel/member-aggr-port # **create br-member-port***breakout-port-num*
7. UCS-A /eth-storage/fabric/port-channel/member-aggr-port/br-member-port # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-storage	Enters Ethernet storage mode.
Step 2	UCS-A# /eth-storage # scope fabric {a b}	Enters Ethernet storage mode for the specified fabric.
Step 3	UCS-A# /eth-storage # scope port-channel <i>port-channel-num</i>	Enters Ethernet storage mode for the specified port-channel.
Step 4	UCS-A# /eth-storage/fabric # enter aggr-interface <i>slot-numaggregate-port-num</i>	Enters the interface for the specified aggregate(main) appliance port.
Step 5	UCS-A /eth-storage/fabric/port-channel # enter member-aggr-port <i>slot-id port-id</i>	Enters the appliance port channel member port.
Step 6	UCS-A /eth-storage/fabric/port-channel/member-aggr-port # create br-member-port <i>breakout-port-num</i>	Creates the appliance port channel member.

	Command or Action	Purpose
Step 7	<p>UCS-A /eth-storage/fabric/port-channel/member-aggr-port/br-member-port # commit-buffer</p> <p>Example:</p> <p>The following example creates an appliance port channel member for an appliance port 2, and commits the transaction:</p> <pre>UCS-A# scope eth-storage UCS-A /eth-storage # scope fabric a UCS-A /eth-storage/fabric # scope port-channel 21 UCS-A /eth-storage/fabric/port-channel # enter member-aggr-port 1 2 UCS-A /eth-storage/fabric/port-channel/member-aggr-port # create br-member-port 2 UCS-A /eth-storage/fabric/port-channel/member-aggr-port/br-member-port* # commit-buffer</pre>	Commits the transaction to the server.

Configuring Breakout FCoE Storage Ports

SUMMARY STEPS

1. UCS-A# **scope fc-storage**
2. UCS-A# /fc-storage **scope fabric {a | b}**
3. UCS-A# /fc-storage/fabric **enter aggr-interface** *slot-numaggregate port-num*
4. UCS-A# /fc-storage/fabric/aggr-interface # **create br-interface br-fcoe** *breakout-port-num*
5. UCS-A# /fc-storage/fabric/aggr-interface/br-interface/br-fcoe # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fc-storage	Enters Fibre Channel storage mode.
Step 2	UCS-A# /fc-storage scope fabric {a b}	Enters Fibre Channel storage mode for the specified fabric.
Step 3	UCS-A# /fc-storage/fabric enter aggr-interface <i>slot-numaggregate port-num</i>	Enter the interface for the specified aggregate(main) Fibre Channel storage port.
Step 4	UCS-A# /fc-storage/fabric/aggr-interface # create br-interface br-fcoe <i>breakout-port-num</i>	Creates an interface for the specified breakout Fibre Channel storage port.
Step 5	<p>UCS-A# /fc-storage/fabric/aggr-interface/br-interface/br-fcoe # commit-buffer</p> <p>Example:</p>	Commits the transaction to the server.

Command or Action	Purpose
<p>The following example creates an interface for a breakout Fibre Channel storage port 1 of the aggregate port 21 on slot 1 of fabric a, and commits the transaction:</p> <pre>UCS-A# scope fc-storage UCS-A /fc-storage # scope fabric a UCS-A /fc-storage/fabric # enter aggr-interface 1 21 UCS-A /fc-storage/fabric/aggr-interface # create br-interface 1 UCS-A /eth-uplink/fabric/aggr-interface/br-interface/br-fcoe # commit-buffer</pre>	

Configuring a Breakout FCoE Uplink Port

SUMMARY STEPS

1. UCS-A# **scope fc-uplink**
2. UCS-A# /fc-uplink **scope fabric {a | b}**
3. UCS-A# /fc-uplink/fabric **enter aggr-interface slot-numaggregate port-num**
4. UCS-A# /fc-uplink/fabric/aggr-interface # **create br-fcoeinterface breakout-port-num**
5. UCS-A# /fc-uplink/fabric/aggr-interface/ br-fcoeinterface # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fc-uplink	Enters FC Uplink mode.
Step 2	UCS-A# /fc-uplink scope fabric {a b}	Enters FC - Uplink mode for the specific fabric.
Step 3	UCS-A# /fc-uplink/fabric enter aggr-interface slot-numaggregate port-num	Enters interface for the specified aggregate(main) FCoE uplink port.
Step 4	UCS-A# /fc-uplink/fabric/aggr-interface # create br-fcoeinterface breakout-port-num	Creates an interface for the specified breakout FCoE uplink port.
Step 5	UCS-A# /fc-uplink/fabric/aggr-interface/ br-fcoeinterface # commit-buffer Example: The following example shows how to create an interface for breakout FCoE uplink port 1 of the aggregate port 20 on slot 1 of fabric A: <pre>UCS-A# scope eth-uplink UCS-A /fc-uplink # scope fabric a UCS-A /fc-uplink/fabric # enter aggr-interface 1 20 UCS-A /fc-uplink/fabric/aggr-interface # create</pre>	Commits the transaction to the server.

	Command or Action	Purpose
	<pre>br-fcoeinterface 1 UCS-A /fc-uplink/fabric/aggr-interface/br-fcoeinterface # commit-buffer</pre>	

Configuring an FCoE Port Channel Member

SUMMARY STEPS

1. UCS-A# **scope fc-uplink**
2. UCS-A# /fc-uplink # **scope fabric {a | b}**
3. UCS-A# /fc-uplink/fabric # **scope fcoe-port-channel fcoe-port-num**
4. UCS-A /fc-uplink/fabric/port-channel # **enter aggr-interface slot-num port-num aggregate-port-num**
5. UCS-A /fc-uplink/fabric/port-channel/member-aggr-port # **create br-member-port breakout-port-num**
6. UCS-A /fc-uplink/fabric/port-channel/member-aggr-port/br-member-port # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fc-uplink	Enters Ethernet storage mode.
Step 2	UCS-A# /fc-uplink # scope fabric {a b}	
Step 3	UCS-A# /fc-uplink/fabric # scope fcoe-port-channel fcoe-port-num	
Step 4	UCS-A /fc-uplink/fabric/port-channel # enter aggr-interface slot-num port-num aggregate-port-num	Enters the FCoE port channel member port.
Step 5	UCS-A /fc-uplink/fabric/port-channel/member-aggr-port # create br-member-port breakout-port-num	Creates the FCoE port channel member for the specified breakout port.
Step 6	<pre>UCS-A /fc-uplink/fabric/port-channel/member-aggr-port/br-member-port # commit-buffer</pre> <p>Example:</p> <p>The following example creates a breakout FCoE port channel member port 4 on aggregate port 21, and commits the transaction:</p> <pre>UCS-A# scope eth-storage UCS-A /fc-uplink # scope fabric a UCS-A /fc-uplink/fabric # scope port-channel 51 UCS-A /fc-uplink/fabric/port-channel # enter member-aggr-port 1 21 UCS-A /fc-uplink/fabric/port-channel/member-aggr-port # create br-member-port 4 UCS-A /fc-uplink/fabric/port-channel/member-aggr-port/br-member-port* # commit-buffer</pre>	Commits the transaction to the server.

Configuring a Breakout VLAN Member Port

SUMMARY STEPS

1. USA-A# **scope eth-uplink**
2. USA-A /eth-uplink # **scope vlan id**
3. USA-A /eth-uplink/vlan # enter member-aggr-port {a|b} **slot-id port id**
4. USA-A /eth-uplink/vlan/member-aggr-port # **create br-member-port breakout-port-name**
5. USA-A /eth-uplink/vlan/member-aggr-port/br-member-port # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	USA-A# scope eth-uplink	Enters Ethernet uplink mode for the specified fabric.
Step 2	USA-A /eth-uplink # scope vlan id	Enters VLAN mode.
Step 3	USA-A /eth-uplink/vlan # enter member-aggr-port {a b} slot-id port id	Enters an interface for the specified fabric, main aggregate port, and subport. breakout VLAN member port.
Step 4	USA-A /eth-uplink/vlan/member-aggr-port # create br-member-port breakout-port-name	Creates an interface for the specified breakout VLAN member port.
Step 5	USA-A /eth-uplink/vlan/member-aggr-port/br-member-port # commit-buffer Example: The following example creates an interface for a VLAN member on the aggregate port 4 on slot 1 of breakout Ethernet uplink port 1, and commits the transaction: <pre> USA-A# scope eth-uplink USA-A /eth-uplink # scope vlan id USA-A /eth-uplink/vlan # enter member-aggr-port a 1 1 USA-A /eth-uplink/vlan/member-aggr-port* # create br-member-port 4 USA-A /eth-uplink/vlan/member-aggr-port/br-member-port* # commit-buffer </pre>	Commits the transaction to the server.

What to do next

Verify that you created the breakout VLAN Member port using the **show** command.

Modifying a Breakout Port

The following table describes how to modify the supported breakout ports.

Breakout Port Type	Scope	CLI Location From Which To Modify	Modify Options
Ethernet Uplink	eth-uplink	UCS-A eth-uplink/fabric/agg-interface # create	mon-src — Creates a monitor source session.
		UCS-A eth-uplink/fabric/agg-interface # set	eth-link-profile — Sets the Ethernet Link profile name. flow-control-policy — Sets the flow control policy that configures the receive and send flow control parameters for the LAN and Ethernet uplink ports. speed — Sets the speed for an Ethernet uplink port. user-label — Assigns an identifying label to the Ethernet Uplink port.
		UCS-A eth-uplink/fabric/agg-interface #	disable — Disables the aggregate interface for the Ethernet Uplink breakout port. enable — Enables the aggregate interface for the Ethernet Uplink breakout port.
Ethernet Uplink port-channel member	fc-storage	UCS-A eth-uplink/fabric/agg-interface # set	eth-link-profile — Sets the Ethernet Link profile name.
		UCS-A eth-uplink/fabric/agg-interface #	disable — Disables the aggregate interface for the breakout Ethernet Uplink port-channel member. enable — Enables the aggregate interface for the breakout Ethernet Uplink port-channel member.

Breakout Port Type	Scope	CLI Location From Which To Modify	Modify Options
FCoE Uplink	fc-uplink	UCS-A /fc-uplink/fabric/aggr-interface/fcoe-interface # create	mon-src — Creates a monitor source session.
		UCS-A /fc-uplink/fabric/aggr-interface/fcoe-interface # set	eth-link-profile — Sets the Ethernet Link profile name. user-label — Assigns an identifying label to the FCoE uplink breakout port.
		UCS-A /fc-uplink/fabric/aggr-interface/fcoe-interface #	disable —Disables the aggregate interface for the FCoE uplink breakout port. enable — Enables the aggregate interface for the FCoE uplink breakout port.
FCoE Uplink port-channel member	eth-uplink	UCS-A /uplink/port-channel/aggr-interface/fcoe-port # set	eth-link-profile — Sets the Ethernet Link profile name.
		A /uplink/port-channel/aggr-interface/fcoe-port #	disable — Disables the aggregate interface for the breakout FCoE uplink port-channel member. enable — Enables the aggregate interface for the breakout FCoE uplink port-channel member.
FCoE Storage port	fc-storage	UCS-A fc-storage/fabric/aggr-interface/br-fcoe # create	mon-src — Creates a monitor source session.
		UCS-A /fc-storage/fabric/aggr-interface/br-fcoe # set	user-label — Assigns an identifying label to the server.
		UCS-A /fc-storage/fabric/aggr-interface/br-fcoe #	disable — Disables the aggregate interface for the breakout FCoE Storage port enable — Enables the aggregate interface for the breakout FCoE Storage port.

Breakout Port Type	Scope	CLI Location From Which To Modify	Modify Options
Appliance Port	eth-storage	UCS-A /eth-storage/fabric-agg-interface # set	<p>adminspeed— Sets the speed for a fabric interface.</p> <p>flowctrlpolicy—Sets the flow control policy that configures the receive and send flow control parameters for the appliance ports.</p> <p>nw-control-policy — Creates a network control policy for the appliance port.</p> <p>pingroupname— Sets the pin group name for the fabric interface.</p> <p>portmode— Sets the appliance port mode.</p> <p>prio — Sets the QoS (Quality of Service) priority level.</p> <p>user-label— Assigns an identifying label to the appliance port.</p>
		UCS-A /eth-storage/fabric-agg-interface # create	<p>eth-target — Creates the Ethernet target endpoint.</p> <p>mon-src— Creates a monitor source session.</p>
		UCS-A /eth-storage/fabric-agg-interface #	<p>disable— Disables the aggregate interface for the appliance breakout port.</p> <p>enable—Enables the aggregate interface for the appliance breakout port.</p>
Appliance port-channel member	eth-storage	UCS-A /eth-storage/port-channel-agg-port #	<p>disable— Disables the aggregate interface for the breakout appliance port-channel member.</p> <p>enable—Enables the aggregate interface for the breakout appliance port-channel member.</p>

Breakout Port Type	Scope	CLI Location From Which To Modify	Modify Options
VLAN Member	eth-uplink	A /eth-uplink/br-aggr-interface # set	isnative — Marks a member-port as a native VLAN.
Pin Group - Pin Target	eth-uplink	N/A	N/A
SPAN (Traffic Monitoring) Destination Port	eth-traffic-mon	A /eth-traffic-mon/br-aggr-interface # set	speed — Sets the speed for the SPAN (Traffic Monitoring) destination port.

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**.
2. UCS-A /eth-uplink # **scope fabric {a | b}**.
3. UCS-A /eth-uplink/fabric # **scope aggr-interface port-number port-id**.
4. UCS-A /eth-uplink/fabric/aggr-interface # **scope br-interface port-id**.
5. UCS-A /eth-uplink/fabric/aggr-interface/br-interface # **create mon-src**.

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink .	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # scope fabric {a b} .	Enters Ethernet uplink fabric mode for the specified fabric.
Step 3	UCS-A /eth-uplink/fabric # scope aggr-interface port-number port-id .	Enters the interface for the specified aggregate(main) Ethernet uplink port.
Step 4	UCS-A /eth-uplink/fabric/aggr-interface # scope br-interface port-id .	Enters the breakout Ethernet port for the specified port number.
Step 5	UCS-A /eth-uplink/fabric/aggr-interface/br-interface # create mon-src . Example: The following example shows how to modify a Ethernet uplink port as a monitor source in breakout port 1 of the aggregate (main) interface in port 1 with an ID of 21. UCS-A# scope eth-uplink UCS-A /eth-uplink # scope fabric a UCS-A /eth-uplink/fabric # scope aggr-interface 1 21 UCS-A /eth-uplink/fabric/aggr-interface # scope br-interface 1 UCS-A /eth-uplink/fabric/aggr-interface/br-interface # create UCS-A	Modifies the interface as a monitoring source.

	Command or Action	Purpose
	/eth-uplink/fabric/aggr-interface/br-interface # create mon-src	

Modifying the Breakout Ethernet Uplink Port Speed and User Label

Enabling or Disabling a Breakout Ethernet Uplink Port

```
pranspat-3gfi-A /eth-uplink/fabric/aggr-interface/br-interface # set
eth-link-profile      Ethernet Link Profile name
flow-control-policy   flow control policy
speed                 Speed
user-label            User Label

pranspat-3gfi-A /eth-uplink/fabric/aggr-interface/br-interface #
disable               Disables services
enable                Enables services
```

Un-configuring Breakout Ports

If you have a breakout on port 2 in slot 1, you can un-configure the breakout port.

Before you begin

You can use the **show port** command to list the ports for the Fabric Interconnect (FI), and select the port that you want to breakout.

SUMMARY STEPS

1. UCS-A# / fabric-interconnect # **show port**
2. UCS-A# **scope cabling**
3. UCS-A# /cabling # **scope fabric {a | b}**
4. UCS-A #/ cabling # **delete breakout {1 | 2}**
5. UCS-A /cabling/fabric/breakout* # **commit** .

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# / fabric-interconnect # show port Example: The following example lists the ports. <pre>Slot Aggr Port Port Oper State Mac Role Xcvr ----- 1 0 1 Link Down 84:B8:02:CA:37:56 Network 1000base T 1 2 1 Sfp Not Present 84:B8:02:CA:37:57 Unknown N/A 1 2 2 Sfp Not Present 84:B8:02:CA:37:57 Unknown N/A 1 2 3 Sfp Not Present</pre>	Displays the ports for the Fabric Interconnect.

	Command or Action	Purpose
	<pre> 84:B8:02:CA:37:57 Unknown N/A 1 2 4 Sfp Not Present 84:B8:02:CA:37:57 Unknown N/A 1 0 3 Sfp Not Present 84:B8:02:CA:37:58 Unknown N/A </pre>	
Step 2	UCS-A# scope cabling	Enters the cabling mode.
Step 3	UCS-A# /cabling # scope fabric {a b}	Specifies fabric a or b.
Step 4	UCS-A #/ cabling # delete breakout {1 2	Warning The breakout port delete action reboots the FI, and any existing configurations on 10G ports are erased.
Step 5	UCS-A /cabling/fabric/breakout* # commit .	Commits the transaction to the system configuration. The FI reboots. After the FI is back up, port 2 in slot one appears as a 40G port.

What to do next

You can use the **show port** to view the unconfigured breakout ports.

Deleting Breakout Ports

You can delete 10 Gig Ethernet breakout ports. Use the **br-interface** or **br-member-port** scopes to select breakout sub-ports 1-4. You must provide the sub-port id for this scope. For example, **scope br-interface sub_port_id**.

The example described in this topic describes how to delete a breakout Ethernet uplink port. The following table describes how to delete the supported Ethernet breakout ports.

Breakout Port Type	Scope	CLI Location From Which To Delete
Ethernet Uplink	eth-uplink	UCS-A /eth-uplink/fabric/aggr-interface # delete br-interface number
Ethernet Uplink port-channel member	eth-uplink	UCS-A /eth-uplink/fabric/port-channel/aggr-interface # delete br-member-port number
FCoE Uplink	fc-uplink	UCS-A /fc-uplink/fabric/aggr-interface # delete br-fcoeinterface number
FCoE Uplink port-channel member	eth-uplink	UCS-A /fc-uplink/fabric/fcoe-port-channel/aggr-interface # delete br-member-port number
FCoE Storage port	fc-storage	UCS-A /fc-storage/fabric/aggr-interface # delete br-interface br-fcoe number
Appliance Port	eth-storage	UCS--A /eth-storage/fabric/port-channel/member-aggr-port # delete br-member-port number

Breakout Port Type	Scope	CLI Location From Which To Delete
Appliance port-channel member	eth-storage	UCS-A /eth-storage/fabric/aggr-interface # delete br-interface <i>number</i>
VLAN Member	eth-uplink	UCS-A /eth-uplink/vlan/member-aggr-port # delete br-member-port <i>number</i>
Pin Group - Pin Target	eth-uplink	UCS-A /eth-uplink/pin-group # delete target <i>number</i>
SPAN (Traffic Monitoring) Destination Port	eth-traffic-mon	UCS-A /eth-traffic-mon/fabric/eth-mon-session/dest-aggr-interface # delete br-dest-interface

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A# /eth-storage # **scope fabric**{a | b}
3. UCS-A /eth-uplink/fabric # **scope port-channel** *number*
4. UCS-A /eth-uplink/fabric/port-channel/aggr-interface # **delete br-member-port** *number*
5. UCS-A /eth-uplink/fabric/port-channel/aggr-interface # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters the Ethernet uplink mode.
Step 2	UCS-A# /eth-storage # scope fabric {a b}	Enters Ethernet storage mode for the specified fabric.
Step 3	UCS-A /eth-uplink/fabric # scope port-channel <i>number</i>	Enters Ethernet uplink fabric port channel mode for the specified port channel.
Step 4	UCS-A /eth-uplink/fabric/port-channel/aggr-interface # delete br-member-port <i>number</i>	Deletes the specified breakout port.
Step 5	UCS-A /eth-uplink/fabric/port-channel/aggr-interface # commit-buffer Example: This example deletes an Ethernet Uplink port-channel member in breakout port 1 of the aggregate (main) interface port 1 slot 1. <pre> UCS-A# scope eth-uplink UCS-A /eth-uplink # scope fabric a UCS-A /eth-uplink/fabric # scope port-channel 1 UCS-A /eth-uplink/fabric/port-channel # enter aggr-interface 1 1 UCS-A /eth-uplink/fabric/port-channel/aggr-interface # delete br-member-port 1 UCS-A </pre>	Commits the transaction to the server.

	Command or Action	Purpose
	<code>/eth-uplink/fabric/port-channel/aggr-interface* # commit-buffer</code>	

What to do next

Verify that you deleted the specified breakout port using the **show** command.

Cisco UCS Mini Scalability Ports

The Cisco UCS 6324 Fabric Interconnect contains a scalability port as well as four unified ports. The scalability port is a 40GB QSFP+ breakout port that, with proper cabling, can support four 1G or 10G SFP+ ports. The scalability ports can be used as a licensed server port for supported Cisco UCS rack servers, an appliance port, or a FCoE port.

In the Cisco UCS Manager GUI, the scalability port is displayed as **Scalability Port 5** below the **Ethernet Ports** node. The individual breakout ports are displayed as **Port 1** through **Port 4**.

In the Cisco UCS Manager CLI, the scalability port is not displayed, but the individual breakout ports are displayed as **Br-Eth1/5/1** through **Br-Eth1/5/4**.

Configuring Scalability Ports

To configure ports, port channel members or SPAN members on the scalability port, scope into the scalability port first, then follow the steps for a standard unified port.

SUMMARY STEPS

1. UCS-A# **scope eth-server**
2. UCS-A /eth-server # **scope fabric {a | b}**
3. UCS-A /eth-server/fabric # **scope aggr-interface slot-num port-num**
4. UCS-A /eth-server/fabric/aggr-interface # **show interface**
5. UCS-A /eth-server/fabric/aggr-interface # **create interface slot-num port-num**
6. UCS-A /eth-server/fabric/aggr-interface # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-server	Enters Ethernet server mode.
Step 2	UCS-A /eth-server # scope fabric {a b}	Enters Ethernet server fabric mode for the specified fabric.
Step 3	UCS-A /eth-server/fabric # scope aggr-interface slot-num port-num	Enters ethernet server fabric aggregate interface mode for the scalability port.
Step 4	UCS-A /eth-server/fabric/aggr-interface # show interface	Displays the interfaces on the scalability port.
Step 5	UCS-A /eth-server/fabric/aggr-interface # create interface slot-num port-num	Creates an interface for the specified Ethernet server port.
Step 6	UCS-A /eth-server/fabric/aggr-interface # commit-buffer	Commits the transaction to the system configuration.

Example

The following example shows how to create an interface for Ethernet server port 3 on the fabric A scalability port and commit the transaction:

```
UCS-A# scope eth-server
UCS-A /eth-server # scope fabric a
UCS-A /eth-server/fabric # scope aggr-interface 1 5
UCS-A /eth-server/fabric/aggr-interface # show interface
Interface:
```

Slot	Id	Aggr-Port	ID	Port	Id	Admin	State	Oper	State	State	Reason
1			5		1	Enabled		Up			
1			5		2	Enabled		Up			
1			5		3	Enabled		Admin	Down	Administratively	Down
1			5		4	Enabled		Admin	Down	Administratively	Down

```
UCS-A /eth-server/fabric/aggr-interface # create interface 1 3
UCS-A /eth-server/fabric/aggr-interface* # commit-buffer
UCS-A /eth-server/fabric/aggr-interface #
```

Beacon LEDs for Unified Ports

Each port on the 6200 series fabric interconnect has a corresponding beacon LED. When the **Beacon LED** property is configured, the beacon LEDs illuminate, showing you which ports are configured in a given port mode.

You can configure the **Beacon LED** property to show you which ports are grouped in one port mode: either Ethernet or Fibre Channel. By default, the Beacon LED property is set to Off.



Note For unified ports on the expansion module, you can reset the **Beacon LED** property to the default value of **Off** during expansion module reboot.

Configuring the Beacon LEDs for Unified Ports

Complete the following task for each module for which you want to configure beacon LEDs.

SUMMARY STEPS

1. UCS-A# **scope fabric-interconnect** {a | b}
2. UCS-A /fabric # **scope card** slot-id
3. UCS-A /fabric/card # **scope beacon-led**
4. UCS-A /fabric/card/beacon-led # **set admin-state** {eth | fc | off}
5. UCS-A /fabric/card/beacon-led # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fabric-interconnect {a b}	Enters fabric interconnect mode for the specified fabric.

	Command or Action	Purpose
Step 2	UCS-A /fabric # scope card <i>slot-id</i>	Enters card mode for the specified fixed or expansion module.
Step 3	UCS-A /fabric/card # scope beacon-led	Enters beacon LED mode.
Step 4	UCS-A /fabric/card/beacon-led # set admin-state {eth fc off}	Specifies which port mode is represented by illuminated beacon LED lights. eth All of the Unified Ports configured in Ethernet mode illuminate. fc All of the Unified Ports configured in Fibre Channel mode illuminate. off Beacon LED lights for all ports on the module are turned off.
Step 5	UCS-A /fabric/card/beacon-led # commit-buffer	Commits the transaction to the system configuration.

Example

The following example illuminates all of the beacon lights for Unified Ports in Ethernet port mode and commits the transaction:

```
UCS-A# scope fabric-interconnect a
UCS-A /fabric # scope card 1
UCS-A /fabric/card # scope beacon-led
UCS-A /fabric/card/beacon-led # set admin-state eth
UCS-A /fabric/card/beacon-led* # commit-buffer
UCS-A /fabric/card/beacon-led #
```

Physical and Backplane Ports

Displaying VIF Port Statistics Obtained From the Adaptor

SUMMARY STEPS

1. UCS-A /fabric-interconnect # **connect nxos** {a | b}
2. UCS-A(nxos)# **show interface vethernet veth-id counters**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A /fabric-interconnect # connect nxos {a b}	Enters NX-OS mode for the fabric interconnect.
Step 2	UCS-A(nxos)# show interface vethernet veth-id counters	Displays VIF port statistics that are obtained from the adaptor.

Example

The following example shows how to display VIF port statistics that are obtained from the adaptor:

```
UCS-A /fabric-interconnect # connect nxos a
UCS-A(nxos) # show interface vethernet 684 counters
```

```
-----
Port                               InOctets                               InUcastPkts
-----
Veth684                             0                                       0

-----
Port                               InMcastPkts                            InBcastPkts
-----
Veth684                             0                                       0

-----
Port                               OutOctets                               OutUcastPkts
-----
Veth684                             0                                       0

-----
Port                               OutMcastPkts                            OutBcastPkts
-----
Veth684                             0                                       0
```

Displaying VIF Port Statistics Obtained From the ASIC

SUMMARY STEPS

1. UCS-A /fabric-interconnect # **connect nxos {a | b}**
2. UCS-A(nxos)# **show platform fwm info lif vethernet veth-id | grep frame**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A /fabric-interconnect # connect nxos {a b}	Enters NX-OS mode for the fabric interconnect.
Step 2	UCS-A(nxos)# show platform fwm info lif vethernet veth-id grep frame	Displays VIF-port RX and TX frame statistics obtained from the ASIC.

	Command or Action	Purpose
		RX statistics are for all type of frames. Tx statistics are only for known unicast frames.

Example

The following example shows how to display VIF-port RX and TX frame statistics obtained from the ASIC:

```
UCS-A /fabric-interconnect # connect nxos a
UCS-A(nxos)# show platform fwm info lif vethernet 684 | grep frame

vif29 pd: rx frames: 0 tx frames: 0;

UCS-A(nxos)#
```

Displaying VIF Ports That Correspond to NIV Ports

SUMMARY STEPS

1. UCS-A /fabric-interconnect # **connect nxos {a | b}**
2. UCS-A(nxos)# **show platform fwm info lif vethernet veth-id | grep niv**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A /fabric-interconnect # connect nxos {a b}	Enters NX-OS mode for the fabric interconnect.
Step 2	UCS-A(nxos)# show platform fwm info lif vethernet veth-id grep niv	Displays VIF ports that correspond to NIV ports.

Example

The following example shows how to display VIF ports that correspond to NIV ports:

```
UCS-A /fabric-interconnect # connect nxos a
UCS-A(nxos)# show platform fwm info lif vethernet 741 | grep niv

vif20 pd: niv_port_id 0x7000001f (the 0x1F or "31" is the Source/Dest-VP index)
```

Verifying Status of Backplane Ports

SUMMARY STEPS

1. UCS-A /fabric-interconnect # **connect nxos {a | b}**
2. UCS-A(nxos)# **show interface br**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A /fabric-interconnect # connect nxos {a b}	Enters NX-OS mode for the fabric interconnect.
Step 2	UCS-A(nxos)# show interface br	Displays the configuration of the interface, including the speed and status of the backplane ports.

Example

The following example shows how to verify the status of backplane ports for fabric interconnect A:

```
UCS-A /fabric-interconnect # connect nxos a
UCS-A(nxos)# show interface br
```

Ethernet Interface	VLAN	Type	Mode	Status	Reason	Speed	Port Ch #
Eth1/1	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/2	1	eth	access	down	SFP not inserted	40G(D)	--
Br-Eth1/3/1	1	eth	access	down	Administratively down	10G(D)	--
Br-Eth1/3/2	1	eth	access	down	Administratively down	10G(D)	--
Br-Eth1/3/3	1	eth	access	down	Administratively down	10G(D)	--
Br-Eth1/3/4	1	eth	access	down	Administratively down	10G(D)	--
Eth1/4	1	eth	access	down	SFP not inserted	40G(D)	--
Br-Eth1/5/1	4044	eth	trunk	down	Link not connected	10G(D)	--
Br-Eth1/5/2	4044	eth	trunk	down	Link not connected	10G(D)	--
Br-Eth1/5/3	4044	eth	trunk	down	Link not connected	10G(D)	--
Br-Eth1/5/4	4044	eth	trunk	down	Link not connected	10G(D)	--
Eth1/6	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/7	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/8	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/9	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/10	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/11	1	eth	fabric	up	none	40G(D)	--
Eth1/12	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/13	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/14	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/15	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/16	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/17	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/18	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/19	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/20	1	eth	access	down	SFP not inserted	40G(D)	--
Br-Eth1/21/1	1	eth	trunk	up	none	10G(D)	--
Br-Eth1/21/2	1	eth	trunk	up	none	10G(D)	--
Br-Eth1/21/3	1	eth	trunk	down	Link not connected	10G(D)	--

Verifying Status of Backplane Ports

Br-Eth1/21/4	1	eth	trunk	up	none	10G(D)	--
Eth1/22	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/23	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/24	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/25	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/26	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/27	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/28	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/29	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/30	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/31	1	eth	access	down	SFP not inserted	40G(D)	--
Eth1/32	1	eth	access	down	SFP not inserted	40G(D)	--

Port-channel Interface	VLAN	Type	Mode	Status	Reason	Speed	Protocol
Pol285	1	eth	vntag	up	none	a-10G(D)	none
Pol286	1	eth	vntag	up	none	a-10G(D)	none
Pol287	1	eth	vntag	up	none	a-10G(D)	none
Pol288	1	eth	vntag	up	none	a-10G(D)	none
Pol289	1	eth	vntag	up	none	a-10G(D)	none

Port	VRF	Status	IP Address	Speed	MTU
mgmt0	--	down	10.197.157.252	--	1500

Vethernet Interface	VLAN	Type	Mode	Status	Reason	Speed
Veth691	4047	virt	trunk	down	nonParticipating	auto
Veth692	4047	virt	trunk	up	none	auto
Veth693	1	virt	trunk	down	nonParticipating	auto
Veth695	1	virt	trunk	up	none	auto
Veth699	1	virt	trunk	up	none	auto

Interface	Secondary VLAN (Type)	Status	Reason
Vlan1	--	down	Administratively down

Ethernet Interface	VLAN	Type	Mode	Status	Reason	Speed	Port Ch #
Eth1/1/1	1	eth	vntag	up	none	10G(D)	1286
Eth1/1/2	1	eth	access	down	Administratively down	10G(D)	--
Eth1/1/3	1	eth	vntag	up	none	10G(D)	1286
Eth1/1/4	1	eth	access	down	Administratively down	10G(D)	--
Eth1/1/5	1	eth	vntag	up	none	10G(D)	1287
Eth1/1/6	1	eth	access	down	Administratively down	10G(D)	--
Eth1/1/7	1	eth	vntag	up	none	10G(D)	1287
Eth1/1/8	1	eth	access	down	Administratively down	10G(D)	--
Eth1/1/9	1	eth	vntag	up	none	10G(D)	1289
Eth1/1/10	1	eth	access	down	Administratively down	10G(D)	--
Eth1/1/11	1	eth	vntag	up	none	10G(D)	1289
Eth1/1/12	1	eth	access	down	Administratively down	10G(D)	--
Eth1/1/13	1	eth	vntag	up	none	10G(D)	1285
Eth1/1/14	1	eth	access	down	Administratively down	10G(D)	--
Eth1/1/15	1	eth	vntag	up	none	10G(D)	1285
Eth1/1/16	1	eth	access	down	Administratively down	10G(D)	--
Eth1/1/17	1	eth	access	down	Administratively down	10G(D)	--
Eth1/1/18	1	eth	vntag	up	none	10G(D)	1288

Eth1/1/19	1	eth	access	down	Administratively down	10G(D) --
Eth1/1/20	1	eth	vntag	up	none	10G(D) 1288
Eth1/1/21	1	eth	access	down	Administratively down	10G(D) --
Eth1/1/22	1	eth	access	down	Administratively down	10G(D) --
Eth1/1/23	1	eth	access	down	Administratively down	10G(D) --
Eth1/1/24	1	eth	access	down	Administratively down	10G(D) --
Eth1/1/25	1	eth	access	down	Administratively down	10G(D) --
Eth1/1/26	1	eth	access	down	Administratively down	10G(D) --
Eth1/1/27	1	eth	access	down	Administratively down	10G(D) --
Eth1/1/28	1	eth	access	down	Administratively down	10G(D) --
Eth1/1/29	1	eth	access	down	Administratively down	10G(D) --
Eth1/1/30	1	eth	access	down	Administratively down	10G(D) --
Eth1/1/31	1	eth	access	down	Administratively down	10G(D) --
Eth1/1/32	1	eth	access	down	Administratively down	10G(D) --
Eth1/1/33	4044	eth	trunk	up	none	1000(D) --

Server Ports

Automatic Configuration of Fabric Interconnect Server Ports

Starting with Cisco UCS Manager release 3.1(3), you can automatically configure the fabric interconnect server ports. The server **Port Auto-Discovery Policy** determines how the system reacts when a new rack server, chassis, or FEX is added. By enabling this policy, Cisco UCS Manager automatically determines the type of device connected to the switch port and configures the switch port accordingly.



Note If you do not want a Cisco UCS C-Series appliance to be UCS Managed, pre-configure the appliance ports before connecting VIC ports to the Cisco UCS fabric interconnect.

Automatically Configuring Server Ports

-
- Step 1** UCS-A# **scope org/**
Enters the root organization mode.
- Step 2** UCS-A / org# **scope por**
Enters organization port discovery policy mode.
- Step 3** UCS-A / org / port-disc-policy# **set descr**
Provides a description for the port discovery policy.
- Step 4** UCS-A / org / port-disc-policy# **set server-auto-disc**
Enables port auto-discovery.

Note By default `server-auto-disc` is disabled. Port auto-discovery is triggered by enabling `server-auto-disc`.

Example

The following example shows how to enable automatic configuration of fabric interconnect server ports:

```
UCS-A# scope org/
UCS-A /org# scope por
UCS-A / org / port-disc-policy # set descr
UCS-A / org / port-disc-policy # set server-auto-disc
```

Configuring a Server Port

All of the port types listed are configurable on both the fixed and expansion module, including server ports, which are not configurable on the 6100 series fabric interconnect expansion module, but are configurable on the 6200 series fabric interconnect expansion module.

SUMMARY STEPS

1. UCS-A# **scope eth-server**
2. UCS-A /eth-server # **scope fabric {a | b}**
3. UCS-A /eth-server/fabric # **create interface slot-num port-num**
4. UCS-A /eth-server/fabric # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-server	Enters Ethernet server mode.
Step 2	UCS-A /eth-server # scope fabric {a b}	Enters Ethernet server fabric mode for the specified fabric.
Step 3	UCS-A /eth-server/fabric # create interface slot-num port-num	Creates an interface for the specified Ethernet server port.
Step 4	UCS-A /eth-server/fabric # commit-buffer	Commits the transaction to the system configuration.

Example

The following example shows how to create an interface for Ethernet server port 4 on slot 1 of fabric B and commit the transaction:

```
UCS-A# scope eth-server
UCS-A /eth-server # scope fabric b
UCS-A /eth-server/fabric # create interface 1 4
UCS-A /eth-server/fabric* # commit-buffer
UCS-A /eth-server/fabric #
```

Unconfiguring a Server Port

SUMMARY STEPS

1. UCS-A# **scope eth-server**
2. UCS-A /eth-server # **scope fabric {a | b}**
3. UCS-A /eth-server/fabric # **delete interface** *slot-num port-num*
4. UCS-A /eth-server/fabric # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-server	Enters Ethernet server mode.
Step 2	UCS-A /eth-server # scope fabric {a b}	Enters Ethernet server fabric mode for the specified fabric.
Step 3	UCS-A /eth-server/fabric # delete interface <i>slot-num port-num</i>	Deletes the interface for the specified Ethernet server port.
Step 4	UCS-A /eth-server/fabric # commit-buffer	Commits the transaction to the system configuration.

Example

The following example unconfigures Ethernet server port 12 on slot 1 of fabric B and commits the transaction:

```
UCS-A# scope eth-server
UCS-A /eth-server # scope fabric b
UCS-A /eth-server/fabric # delete interface 1 12
UCS-A /eth-server/fabric* # commit-buffer
UCS-A /eth-server/fabric #
```

Uplink Ethernet Ports

Configuring an Uplink Ethernet Port

You can configure uplink Ethernet ports on either the fixed module or an expansion module.

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **scope fabric a | b}**
3. UCS-A /eth-uplink/fabric # **create interface** *slot-num port-num*
4. (Optional) UCS-A /eth-uplink/fabric # **set speed {10gbps | 1gbps}**
5. UCS-A /eth-uplink/fabric # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # scope fabric a b	Enters Ethernet uplink fabric mode for the specified fabric.
Step 3	UCS-A /eth-uplink/fabric # create interface slot-num port-num	Creates an interface for the specified Ethernet uplink port.
Step 4	(Optional) UCS-A /eth-uplink/fabric # set speed {10gbps 1gbps}	Sets the speed for the specified Ethernet uplink port. Note For the 6100 series fabric interconnects, the admin speed is only configurable for the first eight ports on a 20-port fabric interconnect and the first 16 ports on a 40-port fabric interconnect.
Step 5	UCS-A /eth-uplink/fabric # commit-buffer	Commits the transaction to the system configuration.

Example

The following example shows how to create an interface for Ethernet uplink port 3 on slot 2 of fabric B, set the speed to 10 gbps, and commit the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope fabric b
UCS-A /eth-uplink/fabric # create interface 2 3
UCS-A /eth-uplink/fabric # set speed 10gbps
UCS-A /eth-uplink/fabric* # commit-buffer
UCS-A /eth-uplink/fabric #
```

Unconfiguring an Uplink Ethernet Port

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **scope fabric {a | b}**
3. UCS-A /eth-uplink/fabric # **delete interface slot-num port-num**
4. UCS-A /eth-uplink/fabric # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # scope fabric {a b}	Enters Ethernet uplink fabric mode for the specified fabric.
Step 3	UCS-A /eth-uplink/fabric # delete interface slot-num port-num	Deletes the interface for the specified Ethernet uplink port.
Step 4	UCS-A /eth-uplink/fabric # commit-buffer	Commits the transaction to the system configuration.

Example

The following example unconfigures Ethernet uplink port 3 on slot 2 of fabric B and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope fabric b
UCS-A /eth-uplink/fabric # delete interface 2 3
UCS-A /eth-uplink/fabric* # commit-buffer
UCS-A /eth-uplink/fabric #
```

Configuring an Uplink Ethernet Port for Forward Error Correction

You can configure forward error correction (FEC) for uplink Ethernet ports, Ethernet appliances, and FCoE uplinks for transceiver modules that operate at 25 Gbps and 100 Gbps speeds that support this feature.

Table 2: FEC CL-74 and FEC CL-91 Support Matrix

Port Speed	FEC CL-74	FEC CL-91
1 Gbps	Not supported	Not supported
10 Gbps	Not supported	Not supported
25 Gbps	Supported	Supported
40 Gbps	Not supported	Not supported
100 Gbps	Not supported	Supported
Auto	Based on inserted transceiver's maximum supported speed	Based on inserted transceiver's maximum supported speed

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **scope fabric a | b**
3. UCS-A /eth-uplink/fabric # **scope interface slot-id port-id**
4. UCS-A /eth-uplink/fabric # **set fec {auto | cl74 | cl91}**
5. UCS-A /eth-uplink/fabric # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # scope fabric a b	Enters Ethernet uplink fabric mode for the specified fabric.
Step 3	UCS-A /eth-uplink/fabric # scope interface slot-id port-id	Enters Ethernet interface mode for the specified interface.

	Command or Action	Purpose
Step 4	Required: UCS-A /eth-uplink/fabric # set fec {auto cl74 cl91}	Sets the forward error correction setting as auto, cl74, or cl91 for the Ethernet uplink port. For the UCS 6454 Fabric Interconnect, forward error correction is only configurable for 25 Gbps or 100 Gbps port speeds.
Step 5	UCS-A /eth-uplink/fabric # commit-buffer	Commits the transaction to the system configuration.

Example

The following example shows how to enable forward error correction cl74 on an interface for Ethernet uplink port 35 on slot 1 of fabric A, and commit the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope fabric a
UCS-A /eth-uplink/fabric # scope interface 1 35
UCS-A /eth-uplink/fabric # set fec cl74
UCS-A /eth-uplink/fabric* # commit-buffer
UCS-A /eth-uplink/fabric #
```

Appliance Ports

Appliance ports are only used to connect fabric interconnects to directly attached NFS storage.



Note

When you create a new appliance VLAN, its IEEE VLAN ID is not added to the LAN Cloud. Therefore, appliance ports that are configured with the new VLAN remain down, by default, due to a pinning failure. To bring up these appliance ports, you have to configure a VLAN in the LAN Cloud with the same IEEE VLAN ID.

Cisco UCS Manager supports up to four appliance ports per fabric interconnect.

Configuring an Appliance Port

You can configure Appliance ports on either the fixed module or an expansion module.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# scope eth-storage	Enters Ethernet storage mode.
Step 2	UCS-A /eth-storage # scope fabric {a b}	Enters Ethernet storage mode for the specified fabric.
Step 3	UCS-A /eth-storage/fabric # create interface slot-num port-num	Creates an interface for the specified appliance port.
Step 4	(Optional) UCS-A /eth-storage/fabric/interface # set portmode {access trunk}	Specifies whether the port mode is access or trunk. By default, the mode is set to trunk.

	Command or Action	Purpose
		Note If traffic for the appliance port needs to traverse the uplink ports, you must also define each VLAN used by this port in the LAN cloud. For example, you need the traffic to traverse the uplink ports if the storage is also used by other servers, or if you want to ensure that traffic fails over to the secondary fabric interconnect if the storage controller for the primary fabric interconnect fails.
Step 5	(Optional) UCS-A /eth-storage/fabric/interface # set pingroupname <i>pin-group name</i>	Specifies the appliance pin target to the specified fabric and port, or fabric and port channel.
Step 6	(Optional) UCS-A /eth-storage/fabric/interface # set prio <i>sys-class-name</i>	<p>Specifies the QoS class for the appliance port. By default, the priority is set to best-effort.</p> <p>The sys-class-name argument can be one of the following class keywords:</p> <ul style="list-style-type: none"> • Fc—Use this priority for QoS policies that control vHBA traffic only. • Platinum—Use this priority for QoS policies that control vNIC traffic only. • Gold—Use this priority for QoS policies that control vNIC traffic only. • Silver—Use this priority for QoS policies that control vNIC traffic only. • Bronze—Use this priority for QoS policies that control vNIC traffic only. • Best Effort—Do not use this priority. It is reserved for the Basic Ethernet traffic lane. If you assign this priority to a QoS policy and configure another system class as CoS 0, Cisco UCS Manager does not default to this system class. It defaults to the priority with CoS 0 for that traffic.
Step 7	(Optional) UCS-A /eth-storage/fabric/interface # set adminspeed {10gbps 1 gbps}	Specifies the admin speed for the interface. By default, the admin speed is set to 10gbps.
Step 8	UCS-A /eth-storage/fabric/interface # commit buffer	Commits the transaction to the system configuration.

Example

The following example creates an interface for an appliance port 2 on slot 3 of fabric B, sets the port mode to access, pins the appliance port to a pin group called pingroup1, sets the QoS class to fc, sets the admin speed to 10 gbps, and commits the transaction:

```

UCS-A# scope eth-storage
UCS-A /eth-storage # scope fabric b
UCS-A /eth-storage/fabric # create interface 3 2
UCS-A /eth-storage/fabric* # set portmode access
UCS-A /eth-storage/fabric* # set pingroupname pingroup1
UCS-A /eth-storage/fabric* # set prio fc
UCS-A /eth-storage/fabric* # set adminspeed 10gbps
UCS-A /eth-storage/fabric* # commit-buffer
UCS-A /eth-storage/fabric #

```

What to do next

Assign a VLAN or target MAC address for the appliance port.

Assigning a Target MAC Address to an Appliance Port or Appliance Port Channel

The following procedure assigns a target MAC address to an appliance port. To assign a target MAC address to an appliance port channel, scope to the port channel instead of the interface.

SUMMARY STEPS

1. UCS-A# **scope eth-storage**
2. UCS-A /eth-storage # **scope fabric {a | b}**
3. UCS-A /eth-storage/fabric # **scope interface slot-id port-id**
4. UCS-A /eth-storage/fabric/interface # **create eth-target eth-target name**
5. UCS-A /eth-storage/fabric/interface/eth-target # **set mac-address mac-address**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-storage	Enters Ethernet storage mode.
Step 2	UCS-A /eth-storage # scope fabric {a b}	Enters Ethernet storage mode for the specified fabric.
Step 3	UCS-A /eth-storage/fabric # scope interface slot-id port-id	Enters Ethernet interface mode for the specified interface. Note To assign a target MAC address to an appliance port channel, use the scope port-channel command instead of scope interface .
Step 4	UCS-A /eth-storage/fabric/interface # create eth-target eth-target name	Specifies the name for the specified MAC address target.
Step 5	UCS-A /eth-storage/fabric/interface/eth-target # set mac-address mac-address	Specifies the MAC address in nn:nn:nn:nn:nn:nn format.

Example

The following example assigns a target MAC address for an appliance device on port 3, slot 2 of fabric B and commits the transaction:

```
UCS-A# scope eth-storage
UCS-A /eth-storage* # scope fabric b
UCS-A /eth-storage/fabric* # scope interface 2 3
UCS-A /eth-storage/fabric/interface* # create eth-target macname
UCS-A /eth-storage/fabric/interface* # set mac-address 01:23:45:67:89:ab
UCS-A /eth-storage/fabric/interface* # commit-buffer
UCS-A /eth-storage/fabric #
```

The following example assigns a target MAC address for appliance devices on port channel 13 of fabric B and commits the transaction:

```
UCS-A# scope eth-storage
UCS-A /eth-storage* # scope fabric b
UCS-A /eth-storage/fabric* # scope port-channel 13
UCS-A /eth-storage/fabric/port-channel* # create eth-target macname
UCS-A /eth-storage/fabric/port-channel* # set mac-address 01:23:45:67:89:ab
UCS-A /eth-storage/fabric/port-channel* # commit-buffer
UCS-A /eth-storage/fabric #
```

Creating an Appliance Port

SUMMARY STEPS

1. UCS-A# **scope eth-storage**
2. UCS-A/eth-storage# **create vlan** *vlan-name* *vlan-id*
3. UCS-A/eth-storage/vlan# **set sharing primary**
4. UCS-A/eth-storage/vlan# **commit buffer**
5. UCS-A/eth-storage# **create vlan** *vlan-name* *vlan-id*
6. UCS-A/eth-storage/vlan# **set sharing community**
7. UCS-A/eth-storage/vlan# **set pubnwnname** *primary* *vlan-name*
8. UCS-A/eth-storage/vlan# **commit buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-storage	Enters Ethernet storage mode.
Step 2	UCS-A/eth-storage# create vlan <i>vlan-name</i> <i>vlan-id</i>	Creates a named VLAN, specifies the VLAN name and VLAN ID, and enters Ethernet storage VLAN mode.
Step 3	UCS-A/eth-storage/vlan# set sharing primary	Saves the changes.
Step 4	UCS-A/eth-storage/vlan# commit buffer	Commits the transaction to the system configuration.
Step 5	UCS-A/eth-storage# create vlan <i>vlan-name</i> <i>vlan-id</i>	Creates a named VLAN, specifies the VLAN name and VLAN ID, and enters Ethernet storage VLAN mode .

	Command or Action	Purpose
Step 6	UCS-A/eth-storage/vlan# set sharing community	Associates the primary VLAN to the secondary VLAN that you are creating.
Step 7	UCS-A/eth-storage/vlan# set pubnwnname <i>primary</i> <i>vlan-name</i>	Specifies the primary VLAN to be associated with this secondary VLAN.
Step 8	UCS-A/eth-storage/vlan# commit buffer	Commits the transaction to the system configuration.

Example

The following example creates an appliance port:

```
UCS-A# scope eth-storage
UCS-A/eth-storage# create vlan PRI600 600
UCS-A/eth-storage/vlan* # set sharing primary
UCS-A/eth-storage/vlan* # commit-buffer
UCS-A/eth-storage # create vlan COM602 602
UCS-A/eth-storage/vlan* # set sharing isolated
UCS-A/eth-storage/vlan* # set pubnwnname PRI600
UCS-A/eth-storage/vlan* # commit-buffer
```

Mapping an Appliance Port to a Community VLAN

SUMMARY STEPS

1. UCS-A# **scope eth-storage**
2. UCS-A/eth-storage# **scope fabric** {*a* | *b*}
3. UCS-A/eth-storage/fabric# **create interface** *slot-num port-num*
4. UCS-A/eth-storage/fabric/interface# **exit**
5. UCS-A/eth-storage/fabric# **exit**
6. UCS-A/eth-storage# **scope vlan** *vlan-name*
7. UCS-A/eth-storage/vlan# **create member-port** *fabric slot-num port-num*
8. UCS-A/eth-storage/vlan/member-port# **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-storage	Enters Ethernet storage mode.
Step 2	UCS-A/eth-storage# scope fabric { <i>a</i> <i>b</i> }	Enters Ethernet storage fabric interconnect mode for the specified fabric interconnect.
Step 3	UCS-A/eth-storage/fabric# create interface <i>slot-num</i> <i>port-num</i>	Creates an interface for the specified Ethernet server port.
Step 4	UCS-A/eth-storage/fabric/interface# exit	Exits from the interface. Note Ensure you commit the transaction after associating with the VLAN.

	Command or Action	Purpose
Step 5	UCS-A/eth-storage/fabric# exit	Exits from the fabric.
Step 6	UCS-A/eth-storage# scope vlan <i>vlan-name</i>	Enters the specified VLAN. Note Ensure community VLAN is created in the appliance cloud.
Step 7	UCS-A/eth-storage/vlan# create member-port <i>fabric slot-num port-num</i>	Creates the member port for the specified fabric, assigns the slot number, and port number and enters member port configuration.
Step 8	UCS-A/eth-storage/vlan/member-port# commit	Commits the transaction to the system configuration.

Example

The following example maps an appliance port to an community VLAN:

```
UCS-A# scope eth-storage
UCS-A/eth-storage# scope fabric a
UCS-A/eth-storage/fabric# create interface 1 22
UCS-A/eth-storage/fabric/interface*# exit
UCS-A/eth-storage/fabric*# exit
UCS-A/eth-storage*# scope vlan COM602
UCS-A/eth-storage/vlan*# create member-port a 1 22
UCS-A/eth-storage/vlan/member-port* commit
```

Unconfiguring an Appliance Port

SUMMARY STEPS

1. UCS-A # **scope eth-storage**
2. UCS-A /eth-storage # **scope fabric** {a | b}
3. UCS-A /eth-storage/fabric # **delete eth-interface** *slot-num port-num*
4. UCS-A /eth-storage/fabric # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A # scope eth-storage	Enters Ethernet storage mode.
Step 2	UCS-A /eth-storage # scope fabric {a b}	Enters Ethernet storage mode for the specified fabric.
Step 3	UCS-A /eth-storage/fabric # delete eth-interface <i>slot-num port-num</i>	Deletes the interface for the specified appliance port.
Step 4	UCS-A /eth-storage/fabric # commit-buffer	Commits the transaction to the system configuration.

Example

The following example unconfigures appliance port 3 on slot 2 of fabric B and commits the transaction:

```
UCS-A# scope eth-storage
UCS-A /eth-storage # scope fabric b
UCS-A /eth-storage/fabric # delete eth-interface 2 3
UCS-A /eth-storage/fabric* # commit-buffer
UCS-A /eth-storage/fabric #
```

Configuring an Appliance Port for Forward Error Correction

You can configure forward error correction (FEC) for appliance ports that operate at 25 Gbps and 100 Gbps speeds that support this feature.

Table 3: FEC CL-74 and FEC CL-91 Support Matrix

Port Speed	FEC CL-74	FEC CL-91
1 Gbps	Not supported	Not supported
10 Gbps	Not supported	Not supported
25 Gbps	Supported	Supported
40 Gbps	Not supported	Not supported
100 Gbps	Not supported	Supported
Auto	Based on inserted transceiver's maximum supported speed	Based on inserted transceiver's maximum supported speed

SUMMARY STEPS

1. UCS-A# **scope eth-storage**
2. UCS-A /eth-storage # **scope fabric a | b**}
3. UCS-A /eth-storage/fabric # **scope interface slot-id port-id**
4. UCS-A /eth-storage/fabric # **set fec {auto |cl74 | cl91}**
5. UCS-A /eth-storage/fabric # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-storage	Enters Ethernet storage mode.
Step 2	UCS-A /eth-storage # scope fabric a b }	Enters Ethernet storage fabric mode for the specified fabric.
Step 3	UCS-A /eth-storage/fabric # scope interface slot-id port-id	Enters Ethernet interface mode for the specified interface.
Step 4	Required: UCS-A /eth-storage/fabric # set fec {auto cl74 cl91}	Sets the forward error correction setting as auto, cl74, or cl91 for the Ethernet appliance port. For the UCS 6400

	Command or Action	Purpose
		Series Fabric Interconnect, forward error correction is only configurable for 25 Gbps or 100 Gbps port speeds.
Step 5	UCS-A /eth-storage/fabric # commit-buffer	Commits the transaction to the system configuration.

Example

The following example shows how to enable forward error correction c174 on an interface for Ethernet appliance port 17 on slot 1 of fabric A, and commit the transaction:

```
UCS-A# scope eth-storage
UCS-A /eth-storage # scope fabric a
UCS-A /eth-storage/fabric # scope interface 1 17
UCS-A /eth-storage/fabric # set fec c174
UCS-A /eth-storage/fabric* # commit-buffer
UCS-A /eth-storage/fabric #
```

FCoE Uplink Ports

FCoE uplink ports are physical Ethernet interfaces between the fabric interconnects and the upstream Ethernet switch, used for carrying FCoE traffic. With this support the same physical Ethernet port can carry both Ethernet traffic and Fibre Channel traffic.

FCoE uplink ports connect to upstream Ethernet switches using the FCoE protocol for Fibre Channel traffic. This allows both the Fibre Channel traffic and Ethernet traffic to flow on the same physical Ethernet link.



Note

FCoE uplinks and unified uplinks enable the multi-hop FCoE feature, by extending the unified fabric up to the distribution layer switch.

You can configure the same Ethernet port as any of the following:

- **FCoE uplink port**—As an FCoE uplink port for only Fibre Channel traffic.
- **Uplink port**—As an Ethernet port for only Ethernet traffic.
- **Unified uplink port**—As a unified uplink port to carry both Ethernet and Fibre Channel traffic.

Configuring a FCoE Uplink Port

All of the port types listed are configurable on both the fixed and expansion module, including server ports, which are not configurable on the 6100 series fabric interconnect expansion module, but are configurable on the 6200 series fabric interconnect expansion module.

SUMMARY STEPS

1. UCS-A# **scope fc-uplink**
2. UCS-A /fc-uplink # **scope fabric {a | b}**

3. UCS-A /fc-uplink/fabric # **create fcoeinterface** *slot-numberport-number*
4. UCS-A /fc-uplink/fabric/fabricinterface # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fc-uplink	Enters FC Uplink mode.
Step 2	UCS-A /fc-uplink # scope fabric {a b}	Enters FC - Uplink mode for the specific fabric.
Step 3	UCS-A /fc-uplink/fabric # create fcoeinterface <i>slot-numberport-number</i>	Creates interface for the specified FCoE uplink port.
Step 4	UCS-A /fc-uplink/fabric/fabricinterface # commit-buffer	Commits the transaction to the system configuration.

Example

The following example creates an interface for FCoE uplink port 8 on slot 1 of fabric A and commits the transaction:

```
UCS-A# scope fc-uplink
UCS-A /fc-uplink # scope fabric a
UCS-A /fc-uplink/fabric # create fcoeinterface 1 8
UCS-A /fc-uplink/fabric/fcoeinterface* # commit-buffer
UCS-A /fc-uplink/fabric/fcoeinterface #
```

Unconfiguring a FCoE Uplink Port

SUMMARY STEPS

1. UCS-A# **scope fc-uplink**
2. UCS-A /fc-uplink # **scope fabric**{a | b}
3. UCS-A /fc-uplink/fabric # **delete fcoeinterface** *slot-numberport-number*
4. UCS-A /fc-uplink/fabric/fabricinterface # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fc-uplink	Enters FC Uplink mode.
Step 2	UCS-A /fc-uplink # scope fabric {a b}	Enters FC - Uplink mode for the specific fabric.
Step 3	UCS-A /fc-uplink/fabric # delete fcoeinterface <i>slot-numberport-number</i>	Deletes the specified interface.
Step 4	UCS-A /fc-uplink/fabric/fabricinterface # commit-buffer	Commits the transaction to the system configuration.

Example

The following example deletes the FCoE uplink interface on port 8 on slot 1 of fabric A and commits the transaction:

```
UCS-A# scope fc-uplink
UCS-A /fc-uplink # scope fabric a
UCS-A /fc-uplink/fabric # delete fcoeinterface 1 8
UCS-A /fc-uplink/fabric/fcoeinterface* # commit-buffer
UCS-A /fc-uplink/fabric/fcoeinterface #
```

Viewing FCoE Uplink Ports

SUMMARY STEPS

1. UCS-A# **scope fc-uplink**
2. UCS-A /fc-uplink # **scope fabric {a | b}**
3. UCS-A /fc-uplink/fabric # **show fcoeinterface**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fc-uplink	Enters FC Uplink mode.
Step 2	UCS-A /fc-uplink # scope fabric {a b}	Enters FC - Uplink mode for the specific fabric.
Step 3	UCS-A /fc-uplink/fabric # show fcoeinterface	Lists the available interfaces.

Example

The following example displays the available FCoE uplink interfaces on fabric A:

```
UCS-A# scope fc-uplink
UCS-A /fc-uplink # scope fabric a
UCS-A /fc-uplink/fabric # show fcoeinterface
FCoE Interface:

Slot Id      Port Id      Admin State Operational State Operational State Reason  Li
c State              Grace Prd
-----
1           26 Enabled      Indeterminate
cense Ok              0

Fcoe Member Port:

Port-channel Slot  Port  Oper State      State Reason
-----
1           1    10 Sfp Not Present Unknown
1           1     3 Sfp Not Present Unknown
1           1     4 Sfp Not Present Unknown
1           1     6 Sfp Not Present Unknown
1           1     8 Sfp Not Present Unknown
```

```

2                               1       7 Sfp Not Present Unknown
UCS-A /fc-uplink/fabric #

```

Configuring FCoE Uplink for Forward Error Correction

You can configure forward error correction (FEC) for FCoE uplinks that operate at 25 Gbps and 100 Gbps speeds that support this feature.

Table 4: FEC CL-74 and FEC CL-91 Support Matrix

Port Speed	FEC CL-74	FEC CL-91
1 Gbps	Not supported	Not supported
10 Gbps	Not supported	Not supported
25 Gbps	Supported	Supported
40 Gbps	Not supported	Not supported
100 Gbps	Not supported	Supported
Auto	Based on inserted transceiver's maximum supported speed	Based on inserted transceiver's maximum supported speed

SUMMARY STEPS

1. UCS-A# **scope fc-uplink**
2. UCS-A /fc-uplink # **scope fabric a | b}**
3. UCS-A /fc-uplink/fabric # **scope fcoeinterface slot-id port-id**
4. UCS-A /fc-uplink/fabric/fcoeinterface # **set fec {auto | cl74 | cl91}**
5. UCS-A /fc-uplink/fabric/fcoeinterface # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fc-uplink	Enters FCoE uplink mode.
Step 2	UCS-A /fc-uplink # scope fabric a b}	Enters fabric mode for the specified fabric.
Step 3	UCS-A /fc-uplink/fabric # scope fcoeinterface slot-id port-id	Enters FCoE interface mode for the specified interface.
Step 4	Required: UCS-A /fc-uplink/fabric/fcoeinterface # set fec {auto cl74 cl91}	Sets the forward error correction setting as auto, cl74, or cl91 for the FCoE uplink. For the UCS 6400 Series Fabric Interconnect, forward error correction is only configurable for 25 Gbps or 100 Gbps port speeds.
Step 5	UCS-A /fc-uplink/fabric/fcoeinterface # commit-buffer	Commits the transaction to the system configuration.

Example

The following example shows how to enable forward error correction c174 on an interface for FCoE uplink 35 on slot 1 of fabric A, and commits the transaction:

```
UCS-A# scope fc-uplink
UCS-A /fc-uplink # scope fabric a
UCS-A /fc-uplink/fabric # scope fcoeinterface 1 35
UCS-A /fc-uplink/fabric/fcoeinterface # set fec c174
UCS-A /fc-uplink/fabric/fcoeinterface # commit-buffer
```

Unified Storage Ports

Unified storage involves configuring the same physical port as both an Ethernet storage interface and an FCoE storage interface. You can configure any appliance port or FCoE storage port as a unified storage port, on either a fixed module or an expansion module. To configure a unified storage port, you must have the fabric interconnect in Fibre Channel switching mode.

In a unified storage port, you can enable or disable individual FCoE storage or appliance interfaces.

- In an unified storage port, if you do not specify a non-default VLAN for the appliance port, the FCoE-storage-native-vlan will be assigned as the native VLAN on the unified storage port. If the appliance port has a non-default native VLAN specified as native VLAN, this will be assigned as the native VLAN for the unified storage port.
- When you enable or disable the appliance interface, the corresponding physical port is enabled or disabled. So when you disable the appliance interface in unified storage, even if the FCoE storage is enabled, it goes down with the physical port.
- When you enable or disable the FCoE storage interface, the corresponding VFC is enabled or disabled. So when the FCoE storage interface is disabled in a unified storage port, the appliance interface will continue to function normally.

Configuring a Unified Storage Port

SUMMARY STEPS

1. UCS-A# **scope eth-storage**
2. UCS-A /eth-storage # **scope fabric{a | b}**
3. UCS-A /eth-storage/fabric # **create interface slot-num port-num**
4. UCS-A /eth-storage/fabric/interface* # **commit buffer**
5. UCS-A /eth-storage/fabric/interface* # **scope fc-storage**
6. UCS-A /fc-storage* # **scope fabric{a | b}**
7. UCS-A /fc-storage/fabric # **create interface fcoe slot-num port-num**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-storage	Enters Ethernet storage mode.

	Command or Action	Purpose
Step 2	UCS-A /eth-storage # scope fabric {a b}	Enters Ethernet storage mode for the specified fabric.
Step 3	UCS-A /eth-storage/fabric # create interface <i>slot-num</i> <i>port-num</i>	Creates an interface for the specified appliance port.
Step 4	UCS-A /eth-storage/fabric/interface* # commit buffer	Commits the transaction to the system configuration.
Step 5	UCS-A /eth-storage/fabric/interface* # scope fc-storage	Enters FC storage mode.
Step 6	UCS-A /fc-storage* # scope fabric {a b}	Enters Ethernet storage mode for the specific appliance port.
Step 7	UCS-A /fc-storage/fabric # create interface fcoe <i>slot-num</i> <i>port-num</i>	Adds FCoE storage port mode on the appliance port mode and creates a unified storage port.

Example

The following example creates an interface for an appliance port 2 on slot 3 of fabric A, adds fc storage to the same port to convert it as a unified port, and commits the transaction:

```
UCS-A# scope eth-storage
UCS-A /eth-storage # scope fabric a
UCS-A /eth-storage/fabric # create interface 3 2
UCS-A /eth-storage/fabric* # commit-buffer
UCS-A /eth-storage/fabric* # scope fc-storage
UCS-A /fc-storage*# scope fabric a
UCS-A /fc-storage/fabric* # create interface fcoe 3 2
UCS-A /fc-storage/fabric* # commit-buffer
UCS-A /fc-storage/fabric*
```

Unified Uplink Ports

When you configure an Ethernet uplink and an FCoE uplink on the same physical Ethernet port, it is called a unified uplink port. You can individually enable or disable either the FCoE or Ethernet interfaces independently.

- Enabling or disabling the FCoE uplink results in the corresponding VFC being enabled or disabled.
- Enabling or disabling an Ethernet uplink results in the corresponding physical port being enabled or disabled.

If you disable an Ethernet uplink, it disables the underlying physical port in a unified uplink. Therefore, even when the FCoE uplink is enabled, the FCoE uplink also goes down. But if you disable an FCoE uplink, only the VFC goes down. If the Ethernet uplink is enabled, it can still function properly in the unified uplink port.

Configuring a Unified Uplink Port

To configure a unified uplink port, you will convert an existing FCoE uplink port as a unified port.

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **scope fabric {a | b}**
3. UCS-A /eth-uplink/fabric # **create interface 15**
4. UCS-A /eth-uplink/fabric/port-channel # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # scope fabric {a b}	Enters Ethernet uplink fabric mode for the specified fabric.
Step 3	UCS-A /eth-uplink/fabric # create interface 15	Converts the FCoE uplink port as a unified port.
Step 4	UCS-A /eth-uplink/fabric/port-channel # commit-buffer	Commits the transaction to the system configuration.

Example

The following example creates a unified uplink port on an existing FCoE port:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope fabric b
UCS-A /eth-uplink/fabric # create interface 1 5
UCS-A /eth-uplink/fabric/interface* # commit-buffer
UCS-A /eth-uplink/interface #
```

FCoE and Fibre Channel Storage Ports

Configuring a Fibre Channel Storage or FCoE Port

SUMMARY STEPS

1. UCS-A# **scope fc-storage**
2. UCS-A /fc-storage # **scope fabric {a | b}**
3. UCS-A /fc-storage/fabric # **create interface {fc | fcoe} slot-num port-num**
4. UCS-A /fc-storage/fabric # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fc-storage	Enters Fibre Channel storage mode.
Step 2	UCS-A /fc-storage # scope fabric {a b}	Enters Fibre Channel storage mode for the specified fabric.
Step 3	UCS-A /fc-storage/fabric # create interface {fc fcoe} slot-num port-num	Creates an interface for the specified Fibre Channel storage port.

	Command or Action	Purpose
Step 4	UCS-A /fc-storage/fabric # commit-buffer	Commits the transaction.

Example

The following example creates an interface for Fibre Channel storage port 10 on slot 2 of fabric A and commits the transaction:

```
UCS-A# scope fc-storage
UCS-A /fc-storage # scope fabric a
UCS-A /fc-storage/fabric* # create interface fc 2 10
UCS-A /fc-storage/fabric # commit-buffer
```

What to do next

Assign a VSAN.

Unconfiguring a Fibre Channel Storage or FCoE Port

SUMMARY STEPS

1. UCS-A# **scope fc-storage**
2. UCS-A /fc-storage # **scope fabric {a | b}**
3. UCS-A /fc-storage/fabric # **delete interface {fc | fcoe} slot-num port-num**
4. UCS-A /fc-storage/fabric # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fc-storage	Enters Fibre Channel storage mode.
Step 2	UCS-A /fc-storage # scope fabric {a b}	Enters Fibre Channel storage mode for the specified fabric.
Step 3	UCS-A /fc-storage/fabric # delete interface {fc fcoe} slot-num port-num	Deletes the interface for the specified Fibre Channel or FCoE storage port.
Step 4	UCS-A /fc-storage/fabric # commit-buffer	Commits the transaction.

Example

The following example unconfigures Fibre Channel storage port 10 on slot 2 of fabric A and commits the transaction:

```
UCS-A# scope fc-storage
UCS-A /fc-storage # scope fabric a
UCS-A /fc-storage/fabric* # delete interface fc 2 10
UCS-A /fc-storage/fabric # commit-buffer
```

Restoring a Fibre Channel Storage Port Back to an Uplink Fibre Channel Port

SUMMARY STEPS

1. UCS-A# **scope fc-uplink**
2. UCS-A /fc-uplink # **scope fabric {a | b}**
3. UCS-A /fc-uplink/fabric # **create interface slot-num port-num**
4. UCS-A /fc-uplink/fabric # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fc-uplink	Enters Fibre Channel uplink mode.
Step 2	UCS-A /fc-uplink # scope fabric {a b}	Enters Fibre Channel uplink mode for the specified fabric.
Step 3	UCS-A /fc-uplink/fabric # create interface slot-num port-num	Creates an interface for the specified Fibre Channel uplink port.
Step 4	UCS-A /fc-uplink/fabric # commit-buffer	Commits the transaction.

Example

The following example creates an interface for Fibre Channel uplink port 10 on slot 2 of fabric A and commits the transaction:

```
UCS-A# scope fc-uplink
UCS-A /fc-uplink # scope fabric a
UCS-A /fc-uplink/fabric* # create interface 2 10
UCS-A /fc-uplink/fabric # commit-buffer
```

Uplink Ethernet Port Channels

An uplink Ethernet port channel allows you to group several physical uplink Ethernet ports (link aggregation) to create one logical Ethernet link to provide fault-tolerance and high-speed connectivity. In Cisco UCS Manager, you create a port channel first and then add uplink Ethernet ports to the port channel. You can add up to 16 uplink Ethernet ports to a port channel.



Important

The state of a configured port changes to unconfigured in the following scenarios:

- The port is deleted or removed from a port channel. The port channel can be of any type, such as, uplink or storage.
- A port channel is deleted.



Note Cisco UCS uses Link Aggregation Control Protocol (LACP), not Port Aggregation Protocol (PAgP), to group the uplink Ethernet ports into a port channel. If the ports on the upstream switch are not configured for LACP, the fabric interconnects treat all ports in an uplink Ethernet port channel as individual ports, and therefore forward packets.

Configuring an Uplink Ethernet Port Channel

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **scope fabric** {a | b }
3. UCS-A /eth-uplink/fabric # **create port-channel** *port-num*
4. (Optional) UCS-A /eth-uplink/fabric/port-channel # {**enable** | **disable**}
5. (Optional) UCS-A /eth-uplink/fabric/port-channel # **set name** *port-chan-name*
6. (Optional) UCS-A /eth-uplink/fabric/port-channel # **set flow-control-policy** *policy-name*
7. UCS-A /eth-uplink/fabric/port-channel # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # scope fabric {a b }	Enters Ethernet uplink fabric mode for the specified fabric.
Step 3	UCS-A /eth-uplink/fabric # create port-channel <i>port-num</i>	Creates a port channel on the specified Ethernet uplink port, and enters Ethernet uplink fabric port channel mode.
Step 4	(Optional) UCS-A /eth-uplink/fabric/port-channel # { enable disable }	Enables or disables the administrative state of the port channel. The port channel is disabled by default.
Step 5	(Optional) UCS-A /eth-uplink/fabric/port-channel # set name <i>port-chan-name</i>	Specifies the name for the port channel.
Step 6	(Optional) UCS-A /eth-uplink/fabric/port-channel # set flow-control-policy <i>policy-name</i>	Assigns the specified flow control policy to the port channel.
Step 7	UCS-A /eth-uplink/fabric/port-channel # commit-buffer	Commits the transaction to the system configuration.

Example

The following example creates a port channel on port 13 of fabric A, sets the name to portchan13a, enables the administrative state, assigns the flow control policy named flow-con-pol432 to the port channel, and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope fabric a
UCS-A /eth-uplink/fabric # create port-channel 13
UCS-A /eth-uplink/fabric/port-channel* # enable
```

```
UCS-A /eth-uplink/fabric/port-channel* # set name portchan13a
UCS-A /eth-uplink/fabric/port-channel* # set flow-control-policy flow-con-pol432
UCS-A /eth-uplink/fabric/port-channel* # commit-buffer
UCS-A /eth-uplink/fabric/port-channel #
```

Unconfiguring an Uplink Ethernet Port Channel

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **scope fabric {a | b }**
3. UCS-A /eth-uplink/fabric # **delete port-channel** *port-num*
4. UCS-A /eth-uplink/fabric # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # scope fabric {a b }	Enters Ethernet uplink fabric mode for the specified fabric.
Step 3	UCS-A /eth-uplink/fabric # delete port-channel <i>port-num</i>	Deletes the port channel on the specified Ethernet uplink port.
Step 4	UCS-A /eth-uplink/fabric # commit-buffer	Commits the transaction to the system configuration.

Example

The following example unconfigures the port channel on port 13 of fabric A and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope fabric a
UCS-A /eth-uplink/fabric # delete port-channel 13
UCS-A /eth-uplink/fabric* # commit-buffer
UCS-A /eth-uplink/fabric #
```

Adding a Member Port to an Uplink Ethernet Port Channel

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **scope fabric {a | b }**
3. UCS-A /eth-uplink/fabric # **scope port-channel** *port-num*
4. UCS-A /eth-uplink/fabric/port-channel # **create member-port** *slot-num port-num*
5. UCS-A /eth-uplink/fabric/port-channel # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # scope fabric {a b }	Enters Ethernet uplink fabric mode for the specified fabric.
Step 3	UCS-A /eth-uplink/fabric # scope port-channel port-num	Enters Ethernet uplink fabric port channel mode for the specified port channel.
Step 4	UCS-A /eth-uplink/fabric/port-channel # create member-port slot-num port-num	Creates the specified member port from the port channel and enters Ethernet uplink fabric port channel member port mode.
Step 5	UCS-A /eth-uplink/fabric/port-channel # commit-buffer	Commits the transaction to the system configuration.

Example

The following example adds the member port on slot 1, port 7 to the port channel on port 13 of fabric A and commits the transaction.

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope fabric a
UCS-A /eth-uplink/fabric # scope port-channel 13
UCS-A /eth-uplink/fabric/port-channel # create member-port 1 7
UCS-A /eth-uplink/fabric/port-channel* # commit-buffer
UCS-A /eth-uplink/fabric/port-channel #
```

Deleting a Member Port from an Uplink Ethernet Port Channel

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **scope fabric {a | b }**
3. UCS-A /eth-uplink/fabric # **scope port-channel port-num**
4. UCS-A /eth-uplink/fabric/port-channel # **delete member-port slot-num port-num**
5. UCS-A /eth-uplink/fabric/port-channel # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # scope fabric {a b }	Enters Ethernet uplink fabric mode for the specified fabric.
Step 3	UCS-A /eth-uplink/fabric # scope port-channel port-num	Enters Ethernet uplink fabric port channel mode for the specified port channel.

	Command or Action	Purpose
Step 4	UCS-A /eth-uplink/fabric/port-channel # delete member-port <i>slot-num port-num</i>	Deletes the specified member port from the port channel.
Step 5	UCS-A /eth-uplink/fabric/port-channel # commit-buffer	Commits the transaction to the system configuration.

Example

The following example deletes a member port from the port channel on port 13 of fabric A and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope fabric a
UCS-A /eth-uplink/fabric # scope port-channel 13
UCS-A /eth-uplink/fabric/port-channel # delete member-port 1 7
UCS-A /eth-uplink/fabric/port-channel* # commit-buffer
UCS-A /eth-uplink/fabric/port-channel #
```

Appliance Port Channels

An appliance port channel allows you to group several physical appliance ports to create one logical Ethernet storage link for the purpose of providing fault-tolerance and high-speed connectivity. In Cisco UCS Manager, you create a port channel first and then add appliance ports to the port channel. You can add up to eight appliance ports to a port channel.

Configuring an Appliance Port Channel

SUMMARY STEPS

1. UCS-A# **scope eth-storage**
2. UCS-A /eth-storage # **scope fabric** {a | b }
3. UCS-A /eth-storage/fabric # **create port-channel** *port-num*
4. (Optional) UCS-A /eth-storage/fabric/port-channel # {**enable** | **disable**}
5. (Optional) UCS-A /eth-storage/fabric/port-channel # **set name** *port-chan-name*
6. (Optional) UCS-A /eth-storage/fabric/port-channel # **set pingroupname** *pin-group name*
7. (Optional) UCS-A /eth-storage/fabric/port-channel # **set portmode** {**access** | **trunk**}
8. (Optional) UCS-A /eth-storage/fabric/port-channel # **set prio** *sys-class-name*
9. (Optional) UCS-A /eth-storage/fabric/port-channel # **set speed** {**1gbps** | **2gbps** | **4gbps** | **8gbps** | **auto**}
10. UCS-A /eth-storage/fabric/port-channel # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-storage	Enters Ethernet storage mode.
Step 2	UCS-A /eth-storage # scope fabric {a b }	Enters Ethernet storage fabric mode for the specified fabric.

	Command or Action	Purpose
Step 3	UCS-A /eth-storage/fabric # create port-channel <i>port-num</i>	Creates a port channel on the specified Ethernet storage port, and enters Ethernet storage fabric port channel mode.
Step 4	(Optional) UCS-A /eth-storage/fabric/port-channel # { enable disable }	Enables or disables the administrative state of the port channel. The port channel is disabled by default.
Step 5	(Optional) UCS-A /eth-storage/fabric/port-channel # set name <i>port-chan-name</i>	Specifies the name for the port channel.
Step 6	(Optional) UCS-A /eth-storage/fabric/port-channel # set pingroupname <i>pin-group name</i>	Specifies the appliance pin target to the specified fabric and port, or fabric and port channel.
Step 7	(Optional) UCS-A /eth-storage/fabric/port-channel # set portmode { access trunk }	Specifies whether the port mode is access or trunk. By default, the mode is set to trunk.
Step 8	(Optional) UCS-A /eth-storage/fabric/port-channel # set prio <i>sys-class-name</i>	<p>Specifies the QoS class for the appliance port. By default, the priority is set to best-effort.</p> <p>The sys-class-name argument can be one of the following class keywords:</p> <ul style="list-style-type: none"> • Fc—Use this priority for QoS policies that control vHBA traffic only. • Platinum—Use this priority for QoS policies that control vNIC traffic only. • Gold—Use this priority for QoS policies that control vNIC traffic only. • Silver—Use this priority for QoS policies that control vNIC traffic only. • Bronze—Use this priority for QoS policies that control vNIC traffic only. • Best Effort—Do not use this priority. It is reserved for the Basic Ethernet traffic lane. If you assign this priority to a QoS policy and configure another system class as CoS 0, Cisco UCS Manager does not default to this system class. It defaults to the priority with CoS 0 for that traffic.
Step 9	(Optional) UCS-A /eth-storage/fabric/port-channel # set speed { 1gbps 2gbps 4gbps 8gbps auto }	Specifies the speed for the port channel.
Step 10	UCS-A /eth-storage/fabric/port-channel # commit-buffer	Commits the transaction to the system configuration.

Example

The following example creates a port channel on port 13 of fabric A and commits the transaction:

```

UCS-A# scope eth-storage
UCS-A /eth-storage # scope fabric a
UCS-A /eth-storage/fabric # create port-channel 13
UCS-A /eth-storage/fabric/port-channel* # enable
UCS-A /eth-storage/fabric/port-channel* # set name portchan13a
UCS-A /eth-storage/fabric/port-channel* # set pingroupname pingroup1
UCS-A /eth-storage/fabric/port-channel* # set portmode access
UCS-A /eth-storage/fabric/port-channel* # set prio fc
UCS-A /eth-storage/fabric/port-channel* # set speed 2gbps
UCS-A /eth-storage/fabric/port-channel* # commit-buffer
UCS-A /eth-storage/fabric/port-channel #

```

Unconfiguring an Appliance Port Channel

SUMMARY STEPS

1. UCS-A# **scope eth-storage**
2. UCS-A /eth-storage # **scope fabric {a | b}**
3. UCS-A /eth-storage/fabric # **delete port-channel** *port-num*
4. UCS-A /eth-storage/fabric # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-storage	Enters Ethernet storage mode.
Step 2	UCS-A /eth-storage # scope fabric {a b}	Enters Ethernet storage fabric mode for the specified fabric.
Step 3	UCS-A /eth-storage/fabric # delete port-channel <i>port-num</i>	Deletes the port channel from the specified Ethernet storage port.
Step 4	UCS-A /eth-storage/fabric # commit-buffer	Commits the transaction to the system configuration.

Example

The following example unconfigures the port channel on port 13 of fabric A and commits the transaction:

```

UCS-A# scope eth-storage
UCS-A /eth-storage # scope fabric a
UCS-A /eth-storage/fabric # delete port-channel 13
UCS-A /eth-storage/fabric* # commit-buffer
UCS-A /eth-storage/fabric #

```

Enabling or Disabling an Appliance Port Channel

SUMMARY STEPS

1. UCS-A# **scope eth-storage**

2. UCS-A /eth-storage # **scope fabric** {a | b }
3. UCS-A /eth-storage/fabric # **scope port-channel** *port-chan-name*
4. UCS-A /eth-storage/fabric/port-channel # {enable | disable }
5. UCS-A /eth-storage/fabric/port-channel # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-storage	Enters Ethernet storage mode.
Step 2	UCS-A /eth-storage # scope fabric {a b }	Enters Ethernet storage mode for the specified fabric.
Step 3	UCS-A /eth-storage/fabric # scope port-channel <i>port-chan-name</i>	Enters Ethernet storage port channel mode.
Step 4	UCS-A /eth-storage/fabric/port-channel # {enable disable }	Enables or disables the administrative state of the port channel. The port channel is disabled by default.
Step 5	UCS-A /eth-storage/fabric/port-channel # commit-buffer	Commits the transaction to the system configuration.

Example

The following example enables port channel 13 on fabric A and commits the transaction:

```
UCS-A# scope eth-storage
UCS-A /eth-storage # scope fabric a
UCS-A /eth-storage/fabric # scope port-channel 13
UCS-A /eth-storage/fabric/port-channel* # enable
UCS-A /eth-storage/fabric/port-channel* # commit-buffer
UCS-A /eth-storage/fabric/port-channel #
```

Adding a Member Port to an Appliance Port Channel

SUMMARY STEPS

1. UCS-A# **scope eth-storage**
2. UCS-A /eth-storage # **scope fabric** {a | b }
3. UCS-A /eth-storage/fabric # **scope port-channel** *port-num*
4. UCS-A /eth-storage/fabric/port-channel # **create member-port** *slot-num port-num*
5. UCS-A /eth-storage/fabric/port-channel # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-storage	Enters Ethernet storage mode.
Step 2	UCS-A /eth-storage # scope fabric {a b }	Enters Ethernet storage fabric mode for the specified fabric.

	Command or Action	Purpose
Step 3	UCS-A /eth-storage/fabric # scope port-channel <i>port-num</i>	Enters Ethernet storage fabric port channel mode for the specified port channel.
Step 4	UCS-A /eth-storage/fabric/port-channel # create member-port <i>slot-num port-num</i>	Creates the specified member port from the port channel and enters Ethernet storage fabric port channel member port mode.
Step 5	UCS-A /eth-storage/fabric/port-channel # commit-buffer	Commits the transaction to the system configuration.

Example

The following example adds the member port on slot 1, port 7 to the port channel on port 13 of fabric A and commits the transaction.

```
UCS-A# scope eth-storage
UCS-A /eth-storage # scope fabric a
UCS-A /eth-storage/fabric # scope port-channel 13
UCS-A /eth-storage/fabric/port-channel # create member-port 1 7
UCS-A /eth-storage/fabric/port-channel* # commit-buffer
UCS-A /eth-storage/fabric/port-channel #
```

Deleting a Member Port from an Appliance Port Channel

SUMMARY STEPS

1. UCS-A# **scope eth-storage**
2. UCS-A /eth-storage # **scope fabric** {a | b }
3. UCS-A /eth-storage/fabric # **scope port-channel** *port-num*
4. UCS-A /eth-storage/fabric/port-channel # **delete member-port** *slot-num port-num*
5. UCS-A /eth-storage/fabric/port-channel # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-storage	Enters Ethernet storage mode.
Step 2	UCS-A /eth-storage # scope fabric {a b }	Enters Ethernet storage fabric mode for the specified fabric.
Step 3	UCS-A /eth-storage/fabric # scope port-channel <i>port-num</i>	Enters Ethernet storage fabric port channel mode for the specified port channel.
Step 4	UCS-A /eth-storage/fabric/port-channel # delete member-port <i>slot-num port-num</i>	Deletes the specified member port from the port channel.
Step 5	UCS-A /eth-storage/fabric/port-channel # commit-buffer	Commits the transaction to the system configuration.

Example

The following example deletes a member port from the port channel on port 13 of fabric A and commits the transaction:

```
UCS-A# scope eth-storage
UCS-A /eth-storage # scope fabric a
UCS-A /eth-storage/fabric # scope port-channel 13
UCS-A /eth-storage/fabric/port-channel # delete member-port 1 7
UCS-A /eth-storage/fabric/port-channel* # commit-buffer
UCS-A /eth-storage/fabric/port-channel #
```

Fibre Channel Port Channels

A Fibre Channel port channel allows you to group several physical Fibre Channel ports (link aggregation) to create one logical Fibre Channel link to provide fault-tolerance and high-speed connectivity. In Cisco UCS Manager, you create a port channel first and then add Fibre Channel ports to the port channel.

**Note**

Fibre Channel port channels are not compatible with non-Cisco technology.

You can create up to four Fibre Channel port channels in each Cisco UCS domain with Cisco UCS 6200, 6300, and 6400 Series Fabric Interconnects. Each Fibre Channel port channel can include a maximum of 16 uplink Fibre Channel ports.

You can create up to two Fibre Channel port channels in each Cisco UCS domain with Cisco UCS 6324 fabric interconnects. Each Fibre Channel port channel can include a maximum of four uplink Fibre Channel ports.

Ensure that the Fibre Channel port channel on the upstream NPIV switch is configured with its channel mode as **active**. If both the member port(s) and peer port(s) do not have the same channel mode configured, the port channel will not come up. When the channel mode is configured as **active**, the member ports initiate port channel protocol negotiation with the peer port(s) regardless of the channel group mode of the peer port. If the peer port, while configured in a channel group, does not support the port channel protocol, or responds with a nonnegotiable status, it defaults to the On mode behavior. The **active** port channel mode allows automatic recovery without explicitly enabling and disabling the port channel member ports at either end.

This example shows how to configure channel mode as active:

```
switch(config)# int po114
switch(config-if)# channel mode active
```

Configuring a Fibre Channel Port Channel

**Note**

If you are connecting two Fibre Channel port channels, the admin speed for both port channels must match for the link to operate. If the admin speed for one or both of the Fibre Channel port channels is set to auto, Cisco UCS adjusts the admin speed automatically.

SUMMARY STEPS

1. UCS-A# **scope fc-uplink**
2. UCS-A /fc-uplink # **scope fabric** {a | b }
3. UCS-A /fc-uplink/fabric # **create port-channel** *port-num*
4. (Optional) UCS-A /fc-uplink/fabric/port-channel # {**enable** | **disable**}
5. (Optional) UCS-A /fc-uplink/fabric/port-channel # **set name** *port-chan-name*
6. (Optional) UCS-A /fc-uplink/fabric/port-channel # **set speed** {1gbps | 2gbps | 4gbps | 8gbps | auto}
7. UCS-A /fc-uplink/fabric/port-channel # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fc-uplink	Enters Fibre Channel uplink mode.
Step 2	UCS-A /fc-uplink # scope fabric {a b }	Enters Fibre Channel uplink fabric mode for the specified fabric.
Step 3	UCS-A /fc-uplink/fabric # create port-channel <i>port-num</i>	Creates a port channel on the specified Fibre Channel uplink port, and enters Fibre Channel uplink fabric port channel mode.
Step 4	(Optional) UCS-A /fc-uplink/fabric/port-channel # { enable disable }	Enables or disables the administrative state of the port channel. The port channel is disabled by default.
Step 5	(Optional) UCS-A /fc-uplink/fabric/port-channel # set name <i>port-chan-name</i>	Specifies the name for the port channel.
Step 6	(Optional) UCS-A /fc-uplink/fabric/port-channel # set speed {1gbps 2gbps 4gbps 8gbps auto}	Specifies the speed for the port channel.
Step 7	UCS-A /fc-uplink/fabric/port-channel # commit-buffer	Commits the transaction to the system configuration.

Example

The following example creates port channel 13 on fabric A, sets the name to portchan13a, enables the administrative state, sets the speed to 2 Gbps, and commits the transaction:

```
UCS-A# scope fc-uplink
UCS-A /fc-uplink # scope fabric a
UCS-A /fc-uplink/fabric # create port-channel 13
UCS-A /fc-uplink/fabric/port-channel* # enable
UCS-A /fc-uplink/fabric/port-channel* # set name portchan13a
UCS-A /fc-uplink/fabric/port-channel* # set speed 2gbps
UCS-A /fc-uplink/fabric/port-channel* # commit-buffer
UCS-A /fc-uplink/fabric/port-channel #
```

Configuring a FCoE Port Channel

SUMMARY STEPS

1. UCS-A# **scope fc-uplink**
2. UCS-A /fc-uplink # **scope fabric {a | b}**
3. UCS-A /fc-uplink/fabric # **create fcoe-port-channel** *number*
4. UCS-A /fc-uplink/fabric/fabricinterface # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fc-uplink	Enters FC Uplink mode.
Step 2	UCS-A /fc-uplink # scope fabric {a b}	Enters FC - Uplink mode for the specific fabric.
Step 3	UCS-A /fc-uplink/fabric # create fcoe-port-channel <i>number</i>	Creates port channel for the specified FCoE uplink port.
Step 4	UCS-A /fc-uplink/fabric/fabricinterface # commit-buffer	Commits the transaction to the system configuration.

Example

The following example creates an interface for FCoE uplink port 1 on slot 4 of fabric A and commits the transaction:

```
UCS-A# scope fc-uplink
UCS-A /fc-uplink # scope fabric a
UCS-A /fc-uplink/fabric # create fcoe-port-channel 4
UCS-A /fc-uplink/fabric/fcoe-port-channel* # commit-buffer
UCS-A /fc-uplink/fabric/fcoe-port-channel #
```

Adding Channel Mode Active To The Upstream NPIV Fibre Channel Port Channel

SUMMARY STEPS

1. UCS-A# **scope fc-uplink**
2. UCS-A /fc-uplink # **scope fabric {a | b}**
3. UCS-A /fc-uplink/fabric # **create port-channel** *port-num*
4. (Optional) UCS-A /fc-uplink/fabric/port-channel # **{enable | disable}**
5. (Optional) UCS-A /fc-uplink/fabric/port-channel # **set name** *port-chan-name*
6. (Optional) UCS-A /fc-uplink/fabric/port-channel # **scope** *port-chan-name*
7. (Optional) UCS-A /fc-uplink/fabric/port-channel # **channel mode {active}**
8. UCS-A /fc-uplink/fabric/port-channel # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fc-uplink	Enters Fibre Channel uplink mode.
Step 2	UCS-A /fc-uplink # scope fabric {a b }	Enters Fibre Channel uplink fabric mode for the specified fabric.
Step 3	UCS-A /fc-uplink/fabric # create port-channel <i>port-num</i>	Creates a port channel on the specified Fibre Channel uplink port, and enters Fibre Channel uplink fabric port channel mode.
Step 4	(Optional) UCS-A /fc-uplink/fabric/port-channel # {enable disable}	Enables or disables the administrative state of the port channel. The port channel is disabled by default.
Step 5	(Optional) UCS-A /fc-uplink/fabric/port-channel # set name <i>port-chan-name</i>	Specifies the name for the port channel.
Step 6	(Optional) UCS-A /fc-uplink/fabric/port-channel # scope <i>port-chan-name</i>	Specifies the name for the port channel.
Step 7	(Optional) UCS-A /fc-uplink/fabric/port-channel # channel mode {active}	Configures the channel-mode active on the upstream NPIV switch.
Step 8	UCS-A /fc-uplink/fabric/port-channel # commit-buffer	Commits the transaction to the system configuration.

Example

The following example enables channel mode to active:

```
UCS-A# scope fc-uplink
UCS-A /fc-uplink # scope fabric a
UCS-A /fc-uplink/fabric # create port-channel 13
UCS-A /fc-uplink/fabric/port-channel* # enable
UCS-A /fc-uplink/fabric/port-channel* # set name portchan13a
UCS-A /fc-uplink/fabric/port-channel* # channel mode active
UCS-A /fc-uplink/fabric/port-channel* # commit-buffer
UCS-A /fc-uplink/fabric/port-channel # exit
UCS-A /fc-uplink/fabric/ # show port-channel database

portchan13a
  Administrative channel mode is active
  Operational channel mode is active

UCS-A /fc-uplink/fabric/ #
```

Enabling or Disabling a Fibre Channel Port Channel

SUMMARY STEPS

1. UCS-A# **scope fc-uplink**
2. UCS-A /fc-uplink # **scope fabric {a | b }**
3. UCS-A /fc-uplink/fabric # **scope port-channel** *port-chan-name*

4. UCS-A /fc-uplink/fabric/port-channel # {enable | disable }

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fc-uplink	Enters Fibre Channel uplink mode.
Step 2	UCS-A /fc-uplink # scope fabric {a b }	Enters Fibre Channel uplink mode for the specified fabric.
Step 3	UCS-A /fc-uplink/fabric # scope port-channel <i>port-chan-name</i>	Enters Fibre Channel uplink port channel mode.
Step 4	UCS-A /fc-uplink/fabric/port-channel # {enable disable } }	Enables or disables the administrative state of the port channel. The port channel is disabled by default.

Example

The following example enables port channel 13 on fabric A and commits the transaction:

```
UCS-A# scope fc-uplink
UCS-A /fc-uplink # scope fabric a
UCS-A /fc-uplink/fabric # scope port-channel 13
UCS-A /fc-uplink/fabric/port-channel* # enable
UCS-A /fc-uplink/fabric/port-channel* # commit-buffer
UCS-A /fc-uplink/fabric/port-channel #
```

Adding a Member Port to a Fibre Channel Port Channel

SUMMARY STEPS

1. UCS-A# **scope fc-uplink**
2. UCS-A /fc-uplink # **scope fabric {a | b }**
3. UCS-A /fc-uplink/fabric # **scope port-channel** *port-num*
4. UCS-A /fc-uplink/fabric/port-channel # **create member-port** *slot-num port-num*
5. UCS-A /fc-uplink/fabric/port-channel # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fc-uplink	Enters Fibre Channel uplink mode.
Step 2	UCS-A /fc-uplink # scope fabric {a b }	Enters Fibre Channel uplink fabric mode for the specified fabric.
Step 3	UCS-A /fc-uplink/fabric # scope port-channel <i>port-num</i>	Enters Fibre Channel uplink fabric port channel mode for the specified port channel.
Step 4	UCS-A /fc-uplink/fabric/port-channel # create member-port <i>slot-num port-num</i>	Creates the specified member port from the port channel and enters Fibre Channel uplink fabric port channel member port mode.

	Command or Action	Purpose
Step 5	UCS-A /fc-uplink/fabric/port-channel # commit-buffer	Commits the transaction to the system configuration.

Example

The following example adds the member port on slot 1, port 7 to port channel 13 on fabric A and commits the transaction.

```
UCS-A# scope fc-uplink
UCS-A /fc-uplink # scope fabric a
UCS-A /fc-uplink/fabric # scope port-channel 13
UCS-A /fc-uplink/fabric # create member-port 1 7
UCS-A /fc-uplink/fabric/port-channel* # commit-buffer
UCS-A /fc-uplink/fabric/port-channel #
```

Deleting a Member Port from a Fibre Channel Port Channel

SUMMARY STEPS

1. UCS-A# **scope fc-uplink**
2. UCS-A /fc-uplink # **scope fabric** {a | b}
3. UCS-A /fc-uplink/fabric # **scope port-channel** *port-num*
4. UCS-A /fc-uplink/fabric/port-channel # **delete member-port** *slot-num port-num*
5. UCS-A /fc-uplink/fabric/port-channel # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fc-uplink	Enters Fibre Channel uplink mode.
Step 2	UCS-A /fc-uplink # scope fabric {a b}	Enters Fibre Channel uplink fabric mode for the specified fabric.
Step 3	UCS-A /fc-uplink/fabric # scope port-channel <i>port-num</i>	Enters Fibre Channel uplink fabric port channel mode for the specified port channel.
Step 4	UCS-A /fc-uplink/fabric/port-channel # delete member-port <i>slot-num port-num</i>	Deletes the specified member port from the port channel.
Step 5	UCS-A /fc-uplink/fabric/port-channel # commit-buffer	Commits the transaction to the system configuration.

Example

The following example deletes a member port from port channel 13 on fabric A and commits the transaction:

```
UCS-A# scope fc-uplink
UCS-A /fc-uplink # scope fabric a
```

```
UCS-A /fc-uplink/fabric # scope port-channel 13
UCS-A /fc-uplink/fabric # delete member-port 1 7
UCS-A /fc-uplink/fabric/port-channel* # commit-buffer
UCS-A /fc-uplink/fabric/port-channel #
```

FCoE Port Channels

An FCoE port channel allows you to group several physical FCoE ports to create one logical FCoE port channel. At a physical level, the FCoE port channel carries FCoE traffic over an Ethernet port channel. So an FCoE port channel with a set of members is essentially an Ethernet port channel with the same members. This Ethernet port channel is used as a physical transport for FCoE traffic.

For each FCoE port channel, Cisco UCS Manager creates a VFC internally and binds it to an Ethernet port channel. FCoE traffic received from the hosts is sent over the VFC the same way as the FCoE traffic is sent over Fibre Channel uplinks.

Configuring a FCoE Port Channel

SUMMARY STEPS

1. UCS-A# **scope fc-uplink**
2. UCS-A /fc-uplink # **scope fabric {a | b}**
3. UCS-A /fc-uplink/fabric # **create fcoe-port-channel number**
4. UCS-A /fc-uplink/fabric/fabricinterface # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fc-uplink	Enters FC Uplink mode.
Step 2	UCS-A /fc-uplink # scope fabric {a b}	Enters FC - Uplink mode for the specific fabric.
Step 3	UCS-A /fc-uplink/fabric # create fcoe-port-channel number	Creates port channel for the specified FCoE uplink port.
Step 4	UCS-A /fc-uplink/fabric/fabricinterface # commit-buffer	Commits the transaction to the system configuration.

Example

The following example creates an interface for FCoE uplink port 1 on slot 4 of fabric A and commits the transaction:

```
UCS-A# scope fc-uplink
UCS-A /fc-uplink # scope fabric a
UCS-A /fc-uplink/fabric # create fcoe-port-channel 4
UCS-A /fc-uplink/fabric/fcoe-port-channel* # commit-buffer
UCS-A /fc-uplink/fabric/fcoe-port-channel #
```

Adding a Member Port to a FCoE Uplink Port Channel

SUMMARY STEPS

1. UCS-A# **scope fc-uplink**
2. UCS-A /fc-uplink # **scope fabric {a | b }**
3. UCS-A /fc-uplink/fabric # **scope fcoe-port-channel ID**
4. UCS-A /fc-uplink/fabric/fcoe-port-channel # **create member-port slot-num port-num**
5. UCS-A /fc-uplink/fabric/fcoe-port-channel # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fc-uplink	Enters Fibre Channel uplink mode.
Step 2	UCS-A /fc-uplink # scope fabric {a b }	Enters Fibre Channel uplink fabric mode for the specified fabric.
Step 3	UCS-A /fc-uplink/fabric # scope fcoe-port-channel ID	Enters FCoE uplink port channel mode for the specified port channel.
Step 4	UCS-A /fc-uplink/fabric/fcoe-port-channel # create member-port slot-num port-num	Creates the specified member port from the port channel and enters FCoE uplink fabric port channel member port mode. Note If the FCoE uplink port channel is a unified uplink port channel, you will get the following message: Warning: if this is a unified port channel then member will be added to the ethernet port channel of the same id as well.
Step 5	UCS-A /fc-uplink/fabric/fcoe-port-channel # commit-buffer	Commits the transaction to the system configuration.

Example

The following example adds the member port on slot 1, port 7 to FCoE port channel 13 on fabric A and commits the transaction.

```
UCS-A# scope fc-uplink
UCS-A /fc-uplink # scope fabric a
UCS-A /fc-uplink/fabric # scope fcoe-port-channel 13
UCS-A /fc-uplink/fabric # create member-port 1 7
UCS-A /fc-uplink/fabric/fcoe-port-channel* # commit-buffer
UCS-A /fc-uplink/fabric/fcoe-port-channel #
```

Unified Uplink Port Channel

When you create an Ethernet port channel and an FCoE port channel with the same ID, it is called a unified uplink port channel. When the unified port channel is created, a physical Ethernet port channel and a VFC are created on the fabric interconnect with the specified members. The physical Ethernet port channel is used to carry both Ethernet and FCoE traffic. The VFC binds FCoE traffic to the Ethernet port channel.

The following rules will apply to the member port sets of the unified uplink port channel:

- The Ethernet port channel and FCoE port channel on the same ID, must have the same set of member ports.
- When you add a member port channel to the Ethernet port channel, Cisco UCS Manager adds the same port channel to FCoE port channel as well. Similarly, adding a member to the FCoE port channel adds the member port to the Ethernet port channel.
- When you delete a member port from one of the port channels, Cisco UCS Manager automatically deletes the member port from the other port channel.

If you disable an Ethernet uplink port channel, it disables the underlying physical port channel in a unified uplink port channel. Therefore, even when the FCoE uplink is enabled, the FCoE uplink port channel also goes down. If you disable an FCoE uplink port channel, only the VFC goes down. If the Ethernet uplink port channel is enabled, it can still function properly in the unified uplink port channel.

Configuring a Unified Uplink Port Channel

To configure a unified uplink port channel, you will convert an existing FCoE uplink port channel as a unified port channel.

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **scope fabric {a | b}**
3. UCS-A /eth-uplink/fabric # **create port-channel ID**
4. UCS-A /eth-uplink/fabric/port-channel # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # scope fabric {a b}	Enters Ethernet uplink fabric mode for the specified fabric.
Step 3	UCS-A /eth-uplink/fabric # create port-channel ID	Creates a port channel for the specified Ethernet uplink port.
Step 4	UCS-A /eth-uplink/fabric/port-channel # commit-buffer	Commits the transaction to the system configuration.

Example

The following example creates a unified uplink port channel on an existing FCoE port channel:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope fabric b
UCS-A /eth-uplink/fabric # create port-channel 2
UCS-A /eth-uplink/fabric/port-channel* # commit-buffer
UCS-A /eth-uplink/fabric #
```

Event Detection and Action

Cisco UCS Manager uses the statistics collection policy to monitor and trigger an alarm when there are faults in the network interface ports connected from the I/O Module (IOM) to the fabric interconnect.

The error statistics for the network interface ports is called NiErrStats and consists of the following errors:

NiErrStats	Description
frameTx	Collects the TX_FRM_ERROR counter values.
tooLong	Collects the RX_TOOLONG counter values.
tooShort	Collects the sum of RX_UNDERSIZE and RX_FRAGMENT counter values.
Crc	Collects the sum of RX_CRERR_NOT_STOMPED and RX_CRCERR_STOMPED counter values.
InRange	Collects the RX_INRANGEERR counter values.



Note

Only active ports collect the network interface port statistics and send the information to Cisco UCS Manager.

Policy-Based Port Error Handling

If Cisco UCS Manager detects any errors on active NI ports, and if the error-disable feature is enabled, Cisco UCS Manager automatically disables the respective FI port that is connected to the NI port that had errors. When a FI port is error disabled, it is effectively shut down and no traffic is sent or received on that port.

The error-disable function serves two purposes:

- It lets you know which FI port is error-disabled and that the connected NI Port has errors.
- It eliminates the possibility that this port can cause other ports, which are connected to the same Chassis/FEX, to fail. Such a failure can occur when the NI port has errors, which can ultimately cause serious network issues. The error-disable function helps prevent these situations.

Creating Threshold Definition

SUMMARY STEPS

1. UCS-A # **scope eth-server**
2. UCS-A/eth-server # **scope stats-threshold-policy default**
3. UCS-A/eth-server/stats-threshold-policy # **create class** *class-name*
4. UCS-A/eth-server/stats-threshold-policy/class # **create property** *property-name*
5. UCS-A/eth-server/stats-threshold-policy/class/property # **set normal-value** *value*
6. UCS-A/eth-server/stats-threshold-policy/class/property # **create threshold-value** {*above-normal* | *below-normal*} {*cleared* | *condition* | *critical* | *info* | *major* | *minor* | *warning*}
7. UCS-A/eth-server/stats-threshold-policy/class/property/threshold-value # **set** {**deescalating** | **escalating**} *value*
8. UCS-A/eth-server/stats-threshold-policy/class/property/threshold-value # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A # scope eth-server	Enters Ethernet storage mode.
Step 2	UCS-A/eth-server # scope stats-threshold-policy default	Enters statistics threshold policy mode.
Step 3	UCS-A/eth-server/stats-threshold-policy # create class <i>class-name</i>	Creates the specified statistics threshold policy class and enters the organization statistics threshold policy class mode. To see a list of the available class name keywords, enter the create class ? command in organization threshold policy mode.
Step 4	UCS-A/eth-server/stats-threshold-policy/class # create property <i>property-name</i>	Creates the specified statistics threshold policy class property and enters the organization statistics threshold policy class property mode. To see a list of the available property name keywords, enter the create property ? command in organization threshold policy class mode.
Step 5	UCS-A/eth-server/stats-threshold-policy/class/property # set normal-value <i>value</i>	Specifies the normal value for the class property. The <i>value</i> format can vary depending on the class property being configured. To see the required format, enter the set normal-value ? command in organization statistics threshold policy class property mode.
Step 6	UCS-A/eth-server/stats-threshold-policy/class/property # create threshold-value { <i>above-normal</i> <i>below-normal</i> } { <i>cleared</i> <i>condition</i> <i>critical</i> <i>info</i> <i>major</i> <i>minor</i> <i>warning</i> }	Creates the specified threshold value for the class property and enters the organization statistics threshold policy class property threshold value mode.
Step 7	UCS-A/eth-server/stats-threshold-policy/class/property/threshold-value # set { deescalating escalating } <i>value</i>	Specifies the deescalating and escalating class property threshold value. The <i>value</i> format can vary depending on the class property threshold value being configured. To see the required format, enter the set deescalating ? or set escalating ? command in the organization statistics threshold policy class property threshold value mode.

	Command or Action	Purpose
Step 8	UCS-A/eth-server/stats-threshold-policy/class/property/threshold-value # commit-buffer	Commits the transaction to the system configuration.

Example

The following example shows how to create a threshold definition:

```
UCS-A # scope eth-server
UCS-A /eth-server # scope stats-threshold-policy default
UCS-A /eth-server/stats-threshold-policy # create class ni-ether-error-stats
UCS-A /eth-server/stats-threshold-policy/class* # create property crc-delta
UCS-A /eth-server/stats-threshold-policy/class/property* # set normal-value 0
UCS-A /eth-server/stats-threshold-policy/class/property* # create threshold-value above-normal
major
UCS-A /eth-server/stats-threshold-policy/class/property/threshold-value* # set escalating
5
UCS-A /eth-server/stats-threshold-policy/class/property/threshold-value* # set deescalating
3
UCS-A /eth-server/stats-threshold-policy/class/property/threshold-value* # commit-buffer
```

Configuring Error Disable on a Fabric Interconnect Port

SUMMARY STEPS

1. UCS-A # **scope eth-server**
2. UCS-A/eth-server # **scope stats-threshold-policy default**
3. UCSA/eth-server/stats-threshold-policy # **scope class** *class-name*
4. UCS-A/eth-server/stats-threshold-policy/class # **scope property** *property-name*
5. UCS-A/eth-server/stats-threshold-policy/class/property # **set error-disable-fi-port** {yes | no}
6. UCS-A/eth-server/stats-threshold-policy/class/property* # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A # scope eth-server	Enters Ethernet storage mode.
Step 2	UCS-A/eth-server # scope stats-threshold-policy default	Enters statistics threshold policy mode.
Step 3	UCSA/eth-server/stats-threshold-policy # scope class <i>class-name</i>	Enters the organization statistics threshold policy class mode for the specified statistics threshold policy class.
Step 4	UCS-A/eth-server/stats-threshold-policy/class # scope property <i>property-name</i>	Enters the organization statistics threshold policy class property mode for the specified statistics threshold policy class property.
Step 5	UCS-A/eth-server/stats-threshold-policy/class/property # set error-disable-fi-port {yes no}	Specifies the error disable state for the class property. Use the no option to disable error disable for the class property.

	Command or Action	Purpose
Step 6	UCS-A/eth-server/stats-threshold-policy/class/property* # commit-buffer	Commits the transaction to the system configuration.

Example

The following example shows how to enable error disable on an FI port:

```
UCS-A # scope eth-server
UCS-A /eth-server # scope stats-threshold-policy default
UCS-A /eth-server/stats-threshold-policy # scope class ni-ether-error-stats
UCS-A /eth-server/stats-threshold-policy/class # scope property crc-delta
UCS-A /eth-server/stats-threshold-policy/class/property # set error-disable-fi-port yes
UCS-A /eth-server/stats-threshold-policy/class/property* # commit-buffer
```

Configuring Auto Recovery on a Fabric Interconnect Port

SUMMARY STEPS

1. UCS-A # **scope eth-server**
2. UCS-A/eth-server # **scope stats-threshold-policy default**
3. UCS-A/eth-server/stats-threshold-policy # **scope class class-name**
4. UCS-A/eth-server/stats-threshold-policy/class # **scope property property-name**
5. UCS-A/eth-server/stats-threshold-policy/class/property # **set auto-recovery {enabled | disabled}**
6. UCS-A/eth-server/stats-threshold-policy/class/property* # **set auto-recovery-time time**
7. UCS-A/eth-server/stats-threshold-policy/class/property* # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A # scope eth-server	Enters Ethernet storage mode.
Step 2	UCS-A/eth-server # scope stats-threshold-policy default	Enters statistics threshold policy mode.
Step 3	UCS-A/eth-server/stats-threshold-policy # scope class class-name	Enters the organization statistics threshold policy class mode for the specified statistics threshold policy class.
Step 4	UCS-A/eth-server/stats-threshold-policy/class # scope property property-name	Enters the organization statistics threshold policy class property mode for the specified statistics threshold policy class property.
Step 5	UCS-A/eth-server/stats-threshold-policy/class/property # set auto-recovery {enabled disabled}	Specifies the auto recovery state for the class property. Use the disabled option to disable auto recovery for the class property.

	Command or Action	Purpose
Step 6	UCS-A/eth-server/stats-threshold-policy/class/property* # set auto-recovery-time <i>time</i>	Specifies the time in minutes after which the port is automatically re-enabled. The auto recovery time can range from 0 minutes to 4294967295 minutes.
Step 7	UCS-A/eth-server/stats-threshold-policy/class/property* # commit-buffer	Commits the transaction to the system configuration.

Example

The following example shows how to configure auto recovery on an FI port:

```
UCS-A # scope eth-server
UCS-A /eth-server # scope stats-threshold-policy default
UCS-A /eth-server/stats-threshold-policy # scope class ni-ether-error-stats
UCS-A /eth-server/stats-threshold-policy/class # scope property crc-delta
UCS-A /eth-server/stats-threshold-policy/class/property # set auto-recovery enabled
UCS-A /eth-server/stats-threshold-policy/class/property* # set auto-recovery-time 5
UCS-A /eth-server/stats-threshold-policy/class/property* # commit-buffer
```

Viewing the Network Interface Port Error Counters

SUMMARY STEPS

1. UCS-A # **scope chassis** *chassis-num*
2. UCS-A/chassis # **scope iom** {a | b}
3. UCS-A/chassis/iom # **scope port-group fabric**
4. UCS-A/chassis/iom/port-group # **scope fabric-if** *fabric-if number*
5. UCS-A/chassis/iom/port-group/fabric-if # **show stats**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A # scope chassis <i>chassis-num</i>	Enters chassis mode for the specified chassis.
Step 2	UCS-A/chassis # scope iom {a b}	Enters chassis IOM mode for the specified IOM.
Step 3	UCS-A/chassis/iom # scope port-group fabric	Enters the network interface port.
Step 4	UCS-A/chassis/iom/port-group # scope fabric-if <i>fabric-if number</i>	Enters the specified network interface port number.
Step 5	UCS-A/chassis/iom/port-group/fabric-if # show stats	Displays the error counters for the network interface port.

Example

The following example shows how to display the statistics for the network interface ports:

```

UCS-A # scope chassis 1
UCS-A/chassis # scope iom a
UCS-A/chassis/iom # scope port-group fabric
UCS-A/chassis/iom/port-group # scope fabric-if 1
UCS-A/chassis/iom/port-group/fabric-if # show stats
NI Ether Error Stats:
Time Collected: 2014-08-20T15:37:24:688
Monitored Object: sys/chassis-1/slot-1/fabric/port-1/ni-err-stats
Suspect: Yes
Crc (errors): 5000
Frame Tx (errors): 0
Too Long (errors): 0
Too Short (errors): 0
In Range (errors): 0
Thresholded: 0

```

Adapter Port Channels

An adapter port channel groups into one logical link all the physical links going from a Cisco UCS Virtual Interface Card (VIC) into an I/O.

Adapter port channels are created and managed internally by Cisco UCS Manager when it detects that the correct hardware is present. Adapter port channels cannot be configured manually. Adapter port channels are viewable using the Cisco UCS Manager GUI or the Cisco UCS Manager CLI.

Viewing Adapter Port Channels

SUMMARY STEPS

1. UCS-A# **scope chassis** *chassis-num*
2. UCS-A /chassis # **scope iom** {a b}
3. UCS-A /chassis/iom # **scope port group**
4. UCS-A /chassis/iom/port group # **show host-port-channel** [detail | expand]

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope chassis <i>chassis-num</i>	Enters chassis mode for the specified chassis.
Step 2	UCS-A /chassis # scope iom {a b}	Enters chassis IOM mode for the specified IOM.
Step 3	UCS-A /chassis/iom # scope port group	Enters port group mode for the specified port group.
Step 4	UCS-A /chassis/iom/port group # show host-port-channel [detail expand]	Displays the adapter port channels on the specified chassis.

Example

This following example shows how to display information on host port channels within a port group mode:

```
UCS-A # scope chassis 1
UCS-A /chassis # scope iom a
UCS-A /chassis/iom # scope port group
UCS-A /chassis/iom/port group # show host-port-channel
```

Host Port channel:

Port Channel Id	Fabric ID	Oper State	State Reason
1289	B	Up	
1290	B	Up	
1306	B	Up	
1307	B	Up	
1309	B	Up	
1315	B	Up	

```
UCS-A /chassis/iom/port group #
```

Fabric Port Channels

Fabric port channels allow you to group several of the physical links from an IOM to a fabric interconnect into one logical link for redundancy and bandwidth sharing. As long as one link in the fabric port channel remains active, the fabric port channel continues to operate.

If the correct hardware is connected, fabric port channels are created by Cisco UCS Manager in the following ways:

- During chassis discovery according to the settings configured in the chassis discovery policy.
- After chassis discovery according to the settings configured in the chassis connectivity policy for a specific chassis.

For each IOM there is a single fabric port channel. Each uplink connecting an IOM to a fabric interconnect can be configured as a discrete link or included in the port channel, but an uplink cannot belong to more than one fabric port channel. For example, if a chassis with two IOMs is discovered and the chassis discovery policy is configured to create fabric port channels, Cisco UCS Manager creates two separate fabric port channels: one for the uplinks connecting IOM-1 and another for the uplinks connecting IOM-2. No other chassis can join these fabric port channels. Similarly, uplinks belonging to the fabric port channel for IOM-1 cannot join the fabric port channel for IOM-2.

Load Balancing Over Ports

Load balancing traffic among ports between IOMs and fabric interconnects uses the following criteria for hashing.

- For Ethernet traffic:
 - Layer 2 source and destination address
 - Layer 3 source and destination address
 - Layer 4 source and destination ports
- For FCoE traffic:
 - Layer 2 source and destination address

Source and destination IDs (SID and DID) and Originator Exchange ID (OXID)

In this example, a 2200 Series IOM module is verified by connecting iom *X* (where *X* is the chassis number).

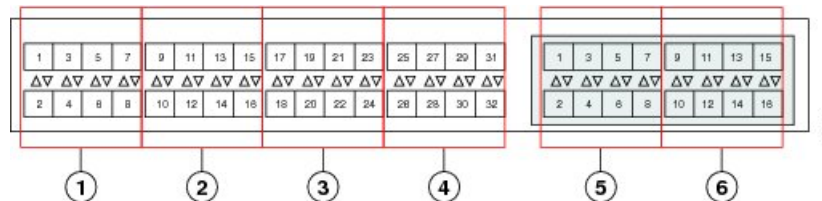
```
show platform software fwmctrl nifport
(....)
Hash Parameters:
  l2_da: 1 l2_sa: 1 l2_vlan: 0
  l3_da: 1 l3_sa: 1
  l4_da: 1 l4_sa: 1
  FCoE l2_da: 1 l2_sa: 1 l2_vlan: 0
  FCoE l3_did: 1 l3_sid: 1 l3_oxid: 1
```

Cabling Considerations for Fabric Port Channels

When you configure the links between the Cisco UCS 2200 Series FEX and a Cisco UCS 6200 series fabric interconnect in fabric port channel mode, the available virtual interface namespace (VIF) on the adapter varies depending on where the FEX uplinks are connected to the fabric interconnect ports.

Inside the 6248 fabric interconnect there are six sets of eight contiguous ports, with each set of ports managed by a single chip. When all uplinks from an FEX are connected to a set of ports managed by a single chip, Cisco UCS Manager maximizes the number of VIFs used in service profiles deployed on the blades in the chassis. If uplink connections from an IOM are distributed across ports managed by separate chips, the VIF count is decreased.

Figure 6: Port Groups for Fabric Port Channels



Caution

Adding a second link to a fabric-port-channel port group is disruptive and will automatically increase the available amount of VIF namespace from 63 to 118. Adding further links is not disruptive and the VIF namespace stays at 118.



Caution

Linking a chassis to two fabric-port-channel port groups does not affect the VIF namespace unless it is manually acknowledged. The VIF namespace is then automatically set to the smaller size fabric port-channel port group usage (either 63 or 118 VIFs) of the two groups.

For high availability cluster-mode applications, we strongly recommend symmetric cabling configurations. If the cabling is asymmetric, the maximum number of VIFs available is the smaller of the two cabling configurations.

For more information on the maximum number of VIFs for your Cisco UCS environment, see the Configuration Limits document for your hardware and software configuration.

Configuring a Fabric Port Channel

SUMMARY STEPS

1. To include all links from the IOM to the fabric interconnect in a fabric port channel during chassis discovery, set the link grouping preference in the chassis discovery policy to port channel.
2. To include links from individual chassis in a fabric port channel during chassis discovery, set the link grouping preference in the chassis connectivity policy to port channel.
3. After chassis discovery, enable or disable additional fabric port channel member ports.

DETAILED STEPS

- | | |
|---------------|---|
| Step 1 | To include all links from the IOM to the fabric interconnect in a fabric port channel during chassis discovery, set the link grouping preference in the chassis discovery policy to port channel. |
| Step 2 | To include links from individual chassis in a fabric port channel during chassis discovery, set the link grouping preference in the chassis connectivity policy to port channel. |
| Step 3 | After chassis discovery, enable or disable additional fabric port channel member ports. |

What to do next

To add or remove chassis links from a fabric port channel after making a change to the chassis discovery policy or the chassis connectivity policy, reacknowledge the chassis. Chassis reacknowledgement is not required to enable or disable chassis member ports from a fabric port channel

Viewing Fabric Port Channels

SUMMARY STEPS

1. UCS-A# **scope eth-server**
2. UCS-A /eth-server # **scope fabric {a | b}**
3. UCS-A /eth-server/fabric # **show fabric-port-channel [detail | expand]**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-server	Enters Ethernet server mode.
Step 2	UCS-A /eth-server # scope fabric {a b}	Enters Ethernet server fabric mode for the specified fabric.
Step 3	UCS-A /eth-server/fabric # show fabric-port-channel [detail expand]	Displays fabric port channels on the specified fabric interconnect.

Example

The following example displays information about configured fabric port channels on fabric interconnect A:

```
UCS-A# scope eth-server
UCS-A /eth-server # scope fabric a
UCS-A /eth-server/fabric # show fabric-port-channel
Fabric Port Channel:
  Port Channel Id Chassis Id Admin State Oper State      State Reason
  -----
          1025 1          Enabled   Failed      No operational members
          1026 2          Enabled    Up
UCS-A /eth-server/fabric #
```

Enabling or Disabling a Fabric Port Channel Member Port

SUMMARY STEPS

1. UCS-A# **scope eth-server**
2. UCS-A /eth-server # **scope fabric {a | b}**
3. UCS-A /eth-server/fabric # **scope fabric-port-channel port-chan-id**
4. UCS-A /eth-server/fabric/fabric-port-channel # **scope member-port slot-id port-id**
5. UCS-A /eth-server/fabric/fabric-port-channel # **{enable | disable}**
6. UCS-A /eth-server/fabric/fabric-port-channel # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-server	Enters Ethernet server mode.
Step 2	UCS-A /eth-server # scope fabric {a b}	Enters Ethernet server fabric mode for the specified fabric.
Step 3	UCS-A /eth-server/fabric # scope fabric-port-channel port-chan-id	Enters Ethernet server fabric, fabric port channel mode for the specified fabric.
Step 4	UCS-A /eth-server/fabric/fabric-port-channel # scope member-port slot-id port-id	Enters Ethernet server fabric, fabric port channel mode for the specified member port.
Step 5	UCS-A /eth-server/fabric/fabric-port-channel # {enable disable}	Enables or disables the specified member port.
Step 6	UCS-A /eth-server/fabric/fabric-port-channel # commit-buffer	Commits the transaction to the system configuration.

Example

The following example disables fabric channel member port 1 31 on fabric port channel 1025 and commits the transaction:

```
UCS-A# scope eth-server
UCS-A /eth-server # scope fabric a
UCS-A /eth-server/fabric # scope fabric-port-channel 1025
UCS-A /eth-server/fabric/fabric-port-channel # scope member-port 1 31
UCS-A /eth-server/fabric/fabric-port-channel/member-port # disable
UCS-A /eth-server/fabric/fabric-port-channel/member-port* # commit-buffer
UCS-A /eth-server/fabric/fabric-port-channel/member-port #
```




CHAPTER 5

VLANs

- [Named VLANs, on page 107](#)
- [Private VLANs, on page 108](#)
- [VLAN Port Limitations, on page 109](#)
- [Configuring Named VLANs, on page 111](#)
- [Configuring Private VLANs, on page 117](#)
- [Community VLANs , on page 125](#)
- [Viewing the VLAN Port Count, on page 129](#)
- [VLAN Port Count Optimization, on page 130](#)
- [VLAN Groups, on page 132](#)
- [VLAN Permissions, on page 137](#)

Named VLANs

A named VLAN creates a connection to a specific external LAN. The VLAN isolates traffic to that external LAN, including broadcast traffic.

The name that you assign to a VLAN ID adds a layer of abstraction that allows you to globally update all servers associated with service profiles that use the named VLAN. You do not need to reconfigure the servers individually to maintain communication with the external LAN.

You can create more than one named VLAN with the same VLAN ID. For example, if servers that host business services for HR and Finance need to access the same external LAN, you can create VLANs named HR and Finance with the same VLAN ID. Then, if the network is reconfigured and Finance is assigned to a different LAN, you only have to change the VLAN ID for the named VLAN for Finance.

In a cluster configuration, you can configure a named VLAN to be accessible only to one fabric interconnect or to both fabric interconnects.

Guidelines for VLAN IDs



Important

VLANs with IDs from 4030 to 4047 and from 4094 to 4095 are reserved. You cannot create VLANs with IDs from this range. Until Cisco UCS Manager Release 4.0(1d), VLAN ID 4093 was in the list of reserved VLANs. VLAN 4093 has been removed from the list of reserved VLANs and is available for configuration.

The VLAN IDs you specify must also be supported on the switch that you are using. For example, on Cisco Nexus 5000 Series switches, the VLAN ID range from 3968 to 4029 is reserved. Before you specify the VLAN IDs in Cisco UCS Manager, make sure that the same VLAN IDs are available on your switch.

VLANs in the LAN cloud and FCoE VLANs in the SAN cloud must have different IDs. Using the same ID for a VLAN and an FCoE VLAN in a VSAN results in a critical fault and traffic disruption for all vNICs and uplink ports using that VLAN. Ethernet traffic is dropped on any VLAN which has an ID that overlaps with an FCoE VLAN ID.

VLAN 4048 is user configurable. However, Cisco UCS Manager uses VLAN 4048 for the following default values. If you want to assign 4048 to a VLAN, you must reconfigure these values:

- After an upgrade to Cisco UCS, Release 2.0—The FCoE storage port native VLAN uses VLAN 4048 by default. If the default FCoE VSAN was set to use VLAN 1 before the upgrade, you must change it to a VLAN ID that is not used or reserved. For example, consider changing the default to 4049 if that VLAN ID is not in use.
- After a fresh install of Cisco UCS, Release 2.0—The FCoE VLAN for the default VSAN uses VLAN 4048 by default. The FCoE storage port native VLAN uses VLAN 4049.

The VLAN name is case sensitive.

Private VLANs

A private VLAN (PVLAN) partitions the Ethernet broadcast domain of a VLAN into subdomains, and allows you to isolate some ports. Each subdomain in a PVLAN includes a primary VLAN and one or more secondary VLANs. All secondary VLANs in a PVLAN must share the same primary VLAN. The secondary VLAN ID differentiates one subdomain from another.

Isolated and Community VLANs

All secondary VLANs in a Cisco UCS domain can be Isolated or Community VLANs.



Note

You cannot configure an isolated VLAN to use with a regular VLAN.

Ports on Isolated VLANs

Communications on an isolated VLAN can only use the associated port in the primary VLAN. These ports are isolated ports and are not configurable in Cisco UCS Manager. A primary VLAN can have only one isolated VLAN, but multiple isolated ports on the same isolated VLAN are allowed. These isolated ports cannot communicate with each other. The isolated ports can communicate only with a regular trunk port or promiscuous port that allows the isolated VLAN.

An isolated port is a host port that belongs to an isolated secondary VLAN. This port has complete isolation from other ports within the same private VLAN domain. PVLANS block all traffic to isolated ports except traffic from promiscuous ports. Traffic received from an isolated port is forwarded only to promiscuous ports. You can have more than one isolated port in a specified isolated VLAN. Each port is completely isolated from all other ports in the isolated VLAN.

Guidelines for Uplink Ports

When you create PVLANS, use the following guidelines:

- The uplink Ethernet port channel cannot be in promiscuous mode.
- Each primary VLAN can have only one isolated VLAN.
- VIFs on VNTAG adapters can have only one isolated VLAN.

Guidelines for VLAN IDs

**Note**

You cannot create VLANs with IDs from 3915 to 4042. These ranges of VLAN IDs are reserved.

The VLAN IDs you specify must also be supported on the switch that you are using. For example, on Cisco Nexus 5000 Series switches, the VLAN ID range from 3968 to 4029 is reserved. Before you specify the VLAN IDs in Cisco UCS Manager, make sure that the same VLAN IDs are available on your switch.

VLANs in the LAN cloud and FCoE VLANs in the SAN cloud must have different IDs. Using the same ID for a VLAN and an FCoE VLAN in a VSAN results in a critical fault and traffic disruption for all vNICs and uplink ports using that VLAN. Ethernet traffic is dropped on any VLAN which has an ID that overlaps with an FCoE VLAN ID.

VLAN 4048 is user configurable. However, Cisco UCS Manager uses VLAN 4048 for the following default values. If you want to assign 4048 to a VLAN, you must reconfigure these values:

- After an upgrade to Cisco UCS, Release 2.0—The FCoE storage port native VLAN uses VLAN 4048 by default. If the default FCoE VSAN was set to use VLAN 1 before the upgrade, you must change it to a VLAN ID that is not used or reserved. For example, consider changing the default to 4049 if that VLAN ID is not in use.
- After a fresh install of Cisco UCS, Release 2.0—The FCoE VLAN for the default VSAN uses VLAN 4048 by default. The FCoE storage port native VLAN uses VLAN 4049.

The VLAN name is case sensitive.

VLAN Port Limitations

Cisco UCS Manager limits the number of VLAN port instances that you can configure under border and server domains on a fabric interconnect.

Types of Ports Included in the VLAN Port Count

The following types of ports are counted in the VLAN port calculation:

- Border uplink Ethernet ports
- Border uplink Ether-channel member ports
- FCoE ports in a SAN cloud
- Ethernet ports in a NAS cloud
- Static and dynamic vNICs created through service profiles
- VM vNICs created as part of a port profile in a hypervisor in hypervisor domain

Based on the number of VLANs configured for these ports, Cisco UCS Manager tracks the cumulative count of VLAN port instances and enforces the VLAN port limit during validation. Cisco UCS Manager reserves some pre-defined VLAN port resources for control traffic. These include management VLANs configured under HIF and NIF ports.

VLAN Port Limit Enforcement

Cisco UCS Manager validates VLAN port availability during the following operations:

- Configuring and unconfiguring border ports and border port channels
- Adding or removing VLANs from a cloud
- Configuring or unconfiguring SAN or NAS ports
- Associating or disassociating service profiles that contain configuration changes
- Configuring or unconfiguring VLANs under vNICs or vHBAs
- Receiving creation or deletion notifications from a VMWare vNIC and from an ESX hypervisor



Note This is outside the control of the Cisco UCS Manager.

- Fabric interconnect reboot
- Cisco UCS Manager upgrade or downgrade

Cisco UCS Manager strictly enforces the VLAN port limit on service profile operations. If Cisco UCS Manager detects that the VLAN port limit is exceeded, the service profile configuration fails during deployment.

Exceeding the VLAN port count in a border domain is less disruptive. When the VLAN port count is exceeded in a border domain Cisco UCS Manager changes the allocation status to Exceeded. To change the status back to **Available**, complete one of the following actions:

- Unconfigure one or more border ports
- Remove VLANs from the LAN cloud
- Unconfigure one or more vNICs or vHBAs

Configuring Named VLANs

Creating a Named VLAN Accessible to Both Fabric Interconnects (Uplink Ethernet Mode)



Important

VLANs with IDs from 4030 to 4047 and from 4094 to 4095 are reserved. You cannot create VLANs with IDs from this range. Until Cisco UCS Manager Release 4.0(1d), VLAN ID 4093 was in the list of reserved VLANs. VLAN 4093 has been removed from the list of reserved VLANs and is available for configuration.

The VLAN IDs you specify must also be supported on the switch that you are using. For example, on Cisco Nexus 5000 Series switches, the VLAN ID range from 3968 to 4029 is reserved. Before you specify the VLAN IDs in Cisco UCS Manager, make sure that the same VLAN IDs are available on your switch.

VLANs in the LAN cloud and FCoE VLANs in the SAN cloud must have different IDs. Using the same ID for a VLAN and an FCoE VLAN in a VSAN results in a critical fault and traffic disruption for all vNICs and uplink ports using that VLAN. Ethernet traffic is dropped on any VLAN which has an ID that overlaps with an FCoE VLAN ID.

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **create vlan** *vlan-name* *vlan-id*
3. UCS-A /eth-uplink/fabric/vlan # **set sharing** {**isolated** | **none** | **primary**}
4. UCS-A /eth-uplink/vlan # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # create vlan <i>vlan-name</i> <i>vlan-id</i>	Creates a named VLAN, specifies the VLAN name and VLAN ID, and enters Ethernet uplink VLAN mode. The VLAN name is case sensitive.
Step 3	UCS-A /eth-uplink/fabric/vlan # set sharing { isolated none primary }	Sets the sharing for the specified VLAN. This can be one of the following: <ul style="list-style-type: none"> • isolated —This is a secondary VLAN associated with a primary VLAN. This VLAN is private. • none —This VLAN does not have any secondary or private VLANs. • primary —This VLAN can have one or more secondary VLANs.

	Command or Action	Purpose
Step 4	UCS-A /eth-uplink/vlan # commit-buffer	Commits the transaction to the system configuration.

Example

The following example creates a named VLAN for both fabric interconnects, names the VLAN accounting, assigns the VLAN ID 2112, sets the sharing to none, and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # create vlan accounting 2112
UCS-A /eth-uplink/vlan* # set sharing none
UCS-A /eth-uplink/vlan* # commit-buffer
UCS-A /eth-uplink/vlan #
```

Creating a Named VLAN Accessible to Both Fabric Interconnects (Ethernet Storage Mode)



Important

VLANs with IDs from 4030 to 4047 and from 4094 to 4095 are reserved. You cannot create VLANs with IDs from this range. Until Cisco UCS Manager Release 4.0(1d), VLAN ID 4093 was in the list of reserved VLANs. VLAN 4093 has been removed from the list of reserved VLANs and is available for configuration.

The VLAN IDs you specify must also be supported on the switch that you are using. For example, on Cisco Nexus 5000 Series switches, the VLAN ID range from 3968 to 4029 is reserved. Before you specify the VLAN IDs in Cisco UCS Manager, make sure that the same VLAN IDs are available on your switch.

VLANs in the LAN cloud and FCoE VLANs in the SAN cloud must have different IDs. Using the same ID for a VLAN and an FCoE VLAN in a VSAN results in a critical fault and traffic disruption for all vNICs and uplink ports using that VLAN. Ethernet traffic is dropped on any VLAN which has an ID that overlaps with an FCoE VLAN ID.

SUMMARY STEPS

1. UCS-A# **scope eth-storage**
2. UCS-A /eth-storage # **create vlan** *vlan-name* *vlan-id*
3. UCS-A /eth-storage/vlan # **create member-port** {a | b} *slot-id* *port-id*
4. UCS-A /eth-storage/vlan/member-port # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-storage	Enters Ethernet storage mode.
Step 2	UCS-A /eth-storage # create vlan <i>vlan-name</i> <i>vlan-id</i>	Creates a named VLAN, specifies the VLAN name and VLAN ID, and enters Ethernet storage VLAN mode. The VLAN name is case sensitive.

	Command or Action	Purpose
Step 3	UCS-A /eth-storage/vlan # create member-port {a b} <i>slot-id port-id</i>	Creates a member port for the specified VLAN on the specified fabric.
Step 4	UCS-A /eth-storage/vlan/member-port # commit-buffer	Commits the transaction to the system configuration.

Example

The following example creates a named VLAN for both fabric interconnects, names the VLAN accounting, assigns the VLAN ID 2112, creates a member port on slot 2, port 20, and commits the transaction:

```
UCS-A# scope eth-storage
UCS-A /eth-storage # create vlan accounting 2112
UCS-A /eth-storage/vlan* # create member-port a 2 20
UCS-A /eth-storage/vlan/member-port* # commit-buffer
UCS-A /eth-storage/vlan/member-port #
```

Creating a Named VLAN Accessible to One Fabric Interconnect (Uplink Ethernet Mode)



Important

VLANs with IDs from 4030 to 4047 and from 4094 to 4095 are reserved. You cannot create VLANs with IDs from this range. Until Cisco UCS Manager Release 4.0(1d), VLAN ID 4093 was in the list of reserved VLANs. VLAN 4093 has been removed from the list of reserved VLANs and is available for configuration.

The VLAN IDs you specify must also be supported on the switch that you are using. For example, on Cisco Nexus 5000 Series switches, the VLAN ID range from 3968 to 4029 is reserved. Before you specify the VLAN IDs in Cisco UCS Manager, make sure that the same VLAN IDs are available on your switch.

VLANs in the LAN cloud and FCoE VLANs in the SAN cloud must have different IDs. Using the same ID for a VLAN and an FCoE VLAN in a VSAN results in a critical fault and traffic disruption for all vNICs and uplink ports using that VLAN. Ethernet traffic is dropped on any VLAN which has an ID that overlaps with an FCoE VLAN ID.

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **scope fabric** {a | b}
3. UCS-A /eth-uplink/fabric # **create vlan** *vlan-name* *vlan-id*
4. UCS-A /eth-uplink/fabric/vlan # **set sharing** {isolated | none | primary}
5. UCS-A /eth-uplink/fabric/vlan # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.

	Command or Action	Purpose
Step 2	UCS-A /eth-uplink # scope fabric {a b}	Enters Ethernet uplink fabric interconnect mode for the specified fabric interconnect (A or B).
Step 3	UCS-A /eth-uplink/fabric # create vlan <i>vlan-name</i> <i>vlan-id</i>	Creates a named VLAN, specifies the VLAN name and VLAN ID, and enters Ethernet uplink fabric interconnect VLAN mode. The VLAN name is case sensitive.
Step 4	UCS-A /eth-uplink/fabric/vlan # set sharing {isolated none primary}	Sets the sharing for the specified VLAN. This can be one of the following: <ul style="list-style-type: none"> • isolated —This is a secondary VLAN associated with a primary VLAN. This VLAN is private. • none —This VLAN does not have any secondary or private VLANs. • primary —This VLAN can have one or more secondary VLANs.
Step 5	UCS-A /eth-uplink/fabric/vlan # commit-buffer	Commits the transaction to the system configuration.

Example

The following example creates a named VLAN for fabric interconnect A, names the VLAN finance, assigns the VLAN ID 3955, sets the sharing to none, and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope fabric a
UCS-A /eth-uplink/fabric # create vlan finance 3955
UCS-A /eth-uplink/fabric/vlan* # set sharing none
UCS-A /eth-uplink/fabric/vlan* # commit-buffer
UCS-A /eth-uplink/fabric/vlan #
```

Creating a Secondary VLAN for a Private VLAN (Accessible to One Fabric Interconnect)



Important

VLANs with IDs from 4030 to 4047 and from 4094 to 4095 are reserved. You cannot create VLANs with IDs from this range. Until Cisco UCS Manager Release 4.0(1d), VLAN ID 4093 was in the list of reserved VLANs. VLAN 4093 has been removed from the list of reserved VLANs and is available for configuration.

The VLAN IDs you specify must also be supported on the switch that you are using. For example, on Cisco Nexus 5000 Series switches, the VLAN ID range from 3968 to 4029 is reserved. Before you specify the VLAN IDs in Cisco UCS Manager, make sure that the same VLAN IDs are available on your switch.

VLANs in the LAN cloud and FCoE VLANs in the SAN cloud must have different IDs. Using the same ID for a VLAN and an FCoE VLAN in a VSAN results in a critical fault and traffic disruption for all vNICs and uplink ports using that VLAN. Ethernet traffic is dropped on any VLAN which has an ID that overlaps with an FCoE VLAN ID.

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **scope fabric {a | b}**
3. UCS-A /eth-uplink/fabric # **create vlan** *vlan-name* *vlan-id*
4. UCS-A /eth-uplink/vlan # **set sharing isolated**
5. UCS-A /eth-uplink/vlan # **set pubnwnname** *primary-vlan-name*
6. UCS-A /eth-uplink/fabric/vlan/member-port # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # scope fabric {a b}	Enters Ethernet uplink fabric interconnect mode for the specified fabric interconnect (A or B).
Step 3	UCS-A /eth-uplink/fabric # create vlan <i>vlan-name</i> <i>vlan-id</i>	Creates a named VLAN, specifies the VLAN name and VLAN ID, and enters Ethernet uplink fabric interconnect VLAN mode. The VLAN name is case sensitive.
Step 4	UCS-A /eth-uplink/vlan # set sharing isolated	Sets the VLAN as the secondary VLAN.
Step 5	UCS-A /eth-uplink/vlan # set pubnwnname <i>primary-vlan-name</i>	Specifies the primary VLAN to be associated with this secondary VLAN.
Step 6	UCS-A /eth-uplink/fabric/vlan/member-port # commit-buffer	Commits the transaction to the system configuration.

Example

The following example creates a named VLAN for fabric interconnect A, names the VLAN finance, assigns the VLAN ID 3955, makes this VLAN the secondary VLAN, associates the secondary VLAN with the primary VLAN, and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope fabric a
UCS-A /eth-uplink/fabric # create vlan finance 3955
UCS-A /eth-uplink/fabric/vlan* # set sharing isolated
UCS-A /eth-uplink/fabric/vlan* # set pubnwnname pvlan1000
UCS-A /eth-uplink/fabric/vlan* # commit-buffer
UCS-A /eth-uplink/fabric/vlan #
```

Deleting a Named VLAN

If Cisco UCS Manager includes a named VLAN with the same VLAN ID as the one you delete, the VLAN is not removed from the fabric interconnect configuration until all named VLANs with that ID are deleted.

If you are deleting a private primary VLAN, ensure that you reassign the secondary VLANs to another working primary VLAN.

Before you begin

Before you delete a VLAN from a fabric interconnect, ensure that the VLAN was removed from all vNICs and vNIC templates.



Note

If you delete a VLAN that is assigned to a vNIC or vNIC template, the vNIC might allow that VLAN to flap.

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. (Optional) UCS-A /eth-uplink # **scope fabric {a | b}**
3. UCS-A /eth-uplink # **delete vlan vlan-name**
4. UCS-A /eth-uplink # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	(Optional) UCS-A /eth-uplink # scope fabric {a b}	Enters Ethernet uplink fabric mode. Use this command when you want to delete a named VLAN only from the specified fabric (a or b).
Step 3	UCS-A /eth-uplink # delete vlan vlan-name	Deletes the specified named VLAN.
Step 4	UCS-A /eth-uplink # commit-buffer	Commits the transaction to the system configuration.

Example

The following example deletes a named VLAN accessible to both fabric interconnects and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # delete vlan accounting
UCS-A /eth-uplink* # commit-buffer
UCS-A /eth-uplink #
```

The following example deletes a named VLAN accessible to one fabric interconnect and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope fabric a
UCS-A /eth-uplink/fabric # delete vlan finance
UCS-A /eth-uplink/fabric* # commit-buffer
UCS-A /eth-uplink/fabric #
```

Configuring Private VLANs

Creating a Primary VLAN for a Private VLAN (Accessible to Both Fabric Interconnects)



Important

VLANs with IDs from 4030 to 4047 and from 4094 to 4095 are reserved. You cannot create VLANs with IDs from this range. Until Cisco UCS Manager Release 4.0(1d), VLAN ID 4093 was in the list of reserved VLANs. VLAN 4093 has been removed from the list of reserved VLANs and is available for configuration.

The VLAN IDs you specify must also be supported on the switch that you are using. For example, on Cisco Nexus 5000 Series switches, the VLAN ID range from 3968 to 4029 is reserved. Before you specify the VLAN IDs in Cisco UCS Manager, make sure that the same VLAN IDs are available on your switch.

VLANs in the LAN cloud and FCoE VLANs in the SAN cloud must have different IDs. Using the same ID for a VLAN and an FCoE VLAN in a VSAN results in a critical fault and traffic disruption for all vNICs and uplink ports using that VLAN. Ethernet traffic is dropped on any VLAN which has an ID that overlaps with an FCoE VLAN ID.

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **create vlan** *vlan-name* *vlan-id*
3. UCS-A /eth-uplink/vlan # **set sharing primary**
4. UCS-A /eth-uplink/vlan # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # create vlan <i>vlan-name</i> <i>vlan-id</i>	Creates a named VLAN, specifies the VLAN name and VLAN ID, and enters Ethernet uplink VLAN mode. The VLAN name is case sensitive.
Step 3	UCS-A /eth-uplink/vlan # set sharing primary	Sets the VLAN as the primary VLAN.
Step 4	UCS-A /eth-uplink/vlan # commit-buffer	Commits the transaction to the system configuration.

Example

The following example creates a named VLAN for both fabric interconnects, names the VLAN accounting, assigns the VLAN ID 2112, makes this VLAN the primary VLAN, and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # create vlan accounting 2112
UCS-A /eth-uplink/vlan* # set sharing primary
UCS-A /eth-uplink/vlan* # commit-buffer
UCS-A /eth-uplink/vlan #
```

Creating a Primary VLAN for a Private VLAN (Accessible to One Fabric Interconnect)

**Important**

VLANs with IDs from 4030 to 4047 and from 4094 to 4095 are reserved. You cannot create VLANs with IDs from this range. Until Cisco UCS Manager Release 4.0(1d), VLAN ID 4093 was in the list of reserved VLANs. VLAN 4093 has been removed from the list of reserved VLANs and is available for configuration.

The VLAN IDs you specify must also be supported on the switch that you are using. For example, on Cisco Nexus 5000 Series switches, the VLAN ID range from 3968 to 4029 is reserved. Before you specify the VLAN IDs in Cisco UCS Manager, make sure that the same VLAN IDs are available on your switch.

VLANs in the LAN cloud and FCoE VLANs in the SAN cloud must have different IDs. Using the same ID for a VLAN and an FCoE VLAN in a VSAN results in a critical fault and traffic disruption for all vNICs and uplink ports using that VLAN. Ethernet traffic is dropped on any VLAN which has an ID that overlaps with an FCoE VLAN ID.

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **scope fabric** {a | b}
3. UCS-A /eth-uplink/fabric # **create vlan** *vlan-name* *vlan-id*
4. UCS-A /eth-uplink/fabric/vlan # **set sharing primary**

5. UCS-A /eth-uplink/fabric/vlan # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # scope fabric {a b}	Enters Ethernet uplink fabric interconnect mode for the specified fabric interconnect.
Step 3	UCS-A /eth-uplink/fabric # create vlan <i>vlan-name</i> <i>vlan-id</i>	Creates a named VLAN, specifies the VLAN name and VLAN ID, and enters Ethernet uplink fabric interconnect VLAN mode. The VLAN name is case sensitive.
Step 4	UCS-A /eth-uplink/fabric/vlan # set sharing primary	Sets the VLAN as the primary VLAN.
Step 5	UCS-A /eth-uplink/fabric/vlan # commit-buffer	Commits the transaction to the system configuration.

Example

The following example creates a named VLAN for fabric interconnect A, names the VLAN finance, assigns the VLAN ID 3955, makes this VLAN the primary VLAN, and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope fabric a
UCS-A /eth-uplink/fabric # create vlan finance 3955
UCS-A /eth-uplink/fabric/vlan* # set sharing primary
UCS-A /eth-uplink/fabric/vlan* # commit-buffer
UCS-A /eth-uplink/fabric/vlan #
```

Creating a Secondary VLAN for a Private VLAN (Accessible to Both Fabric Interconnects)



Important

VLANs with IDs from 4030 to 4047 and from 4094 to 4095 are reserved. You cannot create VLANs with IDs from this range. Until Cisco UCS Manager Release 4.0(1d), VLAN ID 4093 was in the list of reserved VLANs. VLAN 4093 has been removed from the list of reserved VLANs and is available for configuration.

The VLAN IDs you specify must also be supported on the switch that you are using. For example, on Cisco Nexus 5000 Series switches, the VLAN ID range from 3968 to 4029 is reserved. Before you specify the VLAN IDs in Cisco UCS Manager, make sure that the same VLAN IDs are available on your switch.

VLANs in the LAN cloud and FCoE VLANs in the SAN cloud must have different IDs. Using the same ID for a VLAN and an FCoE VLAN in a VSAN results in a critical fault and traffic disruption for all vNICs and uplink ports using that VLAN. Ethernet traffic is dropped on any VLAN which has an ID that overlaps with an FCoE VLAN ID.

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **create vlan** *vlan-name* *vlan-id*
3. UCS-A /eth-uplink/vlan # **set sharing isolated**
4. UCS-A /eth-uplink/vlan # **set pubnwnname** *primary-vlan-name*
5. UCS-A /eth-uplink/vlan # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # create vlan <i>vlan-name</i> <i>vlan-id</i>	Creates a named VLAN, specifies the VLAN name and VLAN ID, and enters Ethernet uplink VLAN mode. The VLAN name is case sensitive.
Step 3	UCS-A /eth-uplink/vlan # set sharing isolated	Sets the VLAN as the secondary VLAN.
Step 4	UCS-A /eth-uplink/vlan # set pubnwnname <i>primary-vlan-name</i>	Specifies the primary VLAN to be associated with this secondary VLAN.
Step 5	UCS-A /eth-uplink/vlan # commit-buffer	Commits the transaction to the system configuration.

Example

The following example creates a named VLAN for both fabric interconnects, names the VLAN accounting, assigns the VLAN ID 2112, makes this VLAN the secondary VLAN, associates the secondary VLAN with the primary VLAN, and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # create vlan accounting 2112
UCS-A /eth-uplink/vlan* # set sharing isolated
UCS-A /eth-uplink/vlan* # set pubnwnname pvlan1000
UCS-A /eth-uplink/vlan* # commit-buffer
UCS-A /eth-uplink/vlan #
```

Creating a Secondary VLAN for a Private VLAN (Accessible to One Fabric Interconnect)



Important

VLANs with IDs from 4030 to 4047 and from 4094 to 4095 are reserved. You cannot create VLANs with IDs from this range. Until Cisco UCS Manager Release 4.0(1d), VLAN ID 4093 was in the list of reserved VLANs. VLAN 4093 has been removed from the list of reserved VLANs and is available for configuration.

The VLAN IDs you specify must also be supported on the switch that you are using. For example, on Cisco Nexus 5000 Series switches, the VLAN ID range from 3968 to 4029 is reserved. Before you specify the VLAN IDs in Cisco UCS Manager, make sure that the same VLAN IDs are available on your switch.

VLANs in the LAN cloud and FCoE VLANs in the SAN cloud must have different IDs. Using the same ID for a VLAN and an FCoE VLAN in a VSAN results in a critical fault and traffic disruption for all vNICs and uplink ports using that VLAN. Ethernet traffic is dropped on any VLAN which has an ID that overlaps with an FCoE VLAN ID.

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **scope fabric {a | b}**
3. UCS-A /eth-uplink/fabric # **create vlan** *vlan-name* *vlan-id*
4. UCS-A /eth-uplink/vlan # **set sharing isolated**
5. UCS-A /eth-uplink/vlan # **set pubnwnname** *primary-vlan-name*
6. UCS-A /eth-uplink/fabric/vlan/member-port # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # scope fabric {a b}	Enters Ethernet uplink fabric interconnect mode for the specified fabric interconnect (A or B).
Step 3	UCS-A /eth-uplink/fabric # create vlan <i>vlan-name</i> <i>vlan-id</i>	Creates a named VLAN, specifies the VLAN name and VLAN ID, and enters Ethernet uplink fabric interconnect VLAN mode. The VLAN name is case sensitive.
Step 4	UCS-A /eth-uplink/vlan # set sharing isolated	Sets the VLAN as the secondary VLAN.
Step 5	UCS-A /eth-uplink/vlan # set pubnwnname <i>primary-vlan-name</i>	Specifies the primary VLAN to be associated with this secondary VLAN.
Step 6	UCS-A /eth-uplink/fabric/vlan/member-port # commit-buffer	Commits the transaction to the system configuration.

Example

The following example creates a named VLAN for fabric interconnect A, names the VLAN finance, assigns the VLAN ID 3955, makes this VLAN the secondary VLAN, associates the secondary VLAN with the primary VLAN, and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope fabric a
UCS-A /eth-uplink/fabric # create vlan finance 3955
UCS-A /eth-uplink/fabric/vlan* # set sharing isolated
UCS-A /eth-uplink/fabric/vlan* # set pubnwnname pvlan1000
UCS-A /eth-uplink/fabric/vlan* # commit-buffer
UCS-A /eth-uplink/fabric/vlan #
```

Allowing PVLANS on vNICs

SUMMARY STEPS

1. UCS-A# **scope org /**
2. UCS-A /org # **scope service-profile** *profile-name*
3. UCS-A /org/service-profile # **scope vnic** *vnic-name*
4. UCS-A /org/service-profile/vnic # **create eth-if** *community-vlan-name*
5. UCS-A /org/service-profile/vnic/eth-if* # **exit**
6. UCS-A /org/service-profile/vnic* # **create eth-if** *primary-vlan-name*
7. UCS-A /org/service-profile/vnic # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope org /	Enters root organization mode.
Step 2	UCS-A /org # scope service-profile <i>profile-name</i>	Commits the transaction to the system configuration.
Step 3	UCS-A /org/service-profile # scope vnic <i>vnic-name</i>	Enters command mode for the specified vNIC.
Step 4	UCS-A /org/service-profile/vnic # create eth-if <i>community-vlan-name</i>	Allows the community VLAN to access the specified vNIC.
Step 5	UCS-A /org/service-profile/vnic/eth-if* # exit	Exits the interface configuration mode for the specified vNIC.
Step 6	UCS-A /org/service-profile/vnic* # create eth-if <i>primary-vlan-name</i>	Allows the primary VLAN to access the specified vNIC.
Step 7	UCS-A /org/service-profile/vnic # commit-buffer	Commits the transaction to the system configuration.

Example

The following example shows how to assign the community VLAN cVLAN102 and the primary VLAN primaryVLAN100 to the vNIC vnic_1 and commits the transaction.

```

UCS-A# scope org /
UCS-A /org # scope service-profile GSP1
UCS-A /org/service-profile # scope vnic vnic_1
UCS-A /org/service-profile/vnic # create eth-if cVLAN102
UCS-A /org/service-profile/vnic/eth-if* # exit
UCS-A /org/service-profile/vnic # create eth-if primaryVLAN100
UCS-A /org/service-profile/vnic* # commit-buffer

```

Creating a Primary VLAN for a Private VLAN on an Appliance Cloud



Important

VLANs with IDs from 4030 to 4047 and from 4094 to 4095 are reserved. You cannot create VLANs with IDs from this range. Until Cisco UCS Manager Release 4.0(1d), VLAN ID 4093 was in the list of reserved VLANs. VLAN 4093 has been removed from the list of reserved VLANs and is available for configuration.

The VLAN IDs you specify must also be supported on the switch that you are using. For example, on Cisco Nexus 5000 Series switches, the VLAN ID range from 3968 to 4029 is reserved. Before you specify the VLAN IDs in Cisco UCS Manager, make sure that the same VLAN IDs are available on your switch.

VLANs in the LAN cloud and FCoE VLANs in the SAN cloud must have different IDs. Using the same ID for a VLAN and an FCoE VLAN in a VSAN results in a critical fault and traffic disruption for all vNICs and uplink ports using that VLAN. Ethernet traffic is dropped on any VLAN which has an ID that overlaps with an FCoE VLAN ID.

SUMMARY STEPS

1. UCS-A# **scope eth-storage**
2. UCS-A /eth-storage # **create vlan** *vlan-name* *vlan-id*
3. UCS-A /eth-storage/vlan* # **set sharing primary**
4. UCS-A /eth-storage/vlan* # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-storage	Enters Ethernet storage mode.
Step 2	UCS-A /eth-storage # create vlan <i>vlan-name</i> <i>vlan-id</i>	Creates a named VLAN, specifies the VLAN name and VLAN ID, and enters Ethernet storage VLAN mode. The VLAN name is case sensitive.
Step 3	UCS-A /eth-storage/vlan* # set sharing primary	Sets the VLAN as the primary VLAN.
Step 4	UCS-A /eth-storage/vlan* # commit-buffer	Commits the transaction to the system configuration.

Example

The following example creates a named VLAN for fabric interconnect A, names the VLAN, assigns the VLAN ID, makes this VLAN the primary VLAN, and commits the transaction:

```

UCS-A# scope eth-storage
UCS-A /eth-storage # create vlan primaryvlan500 500
UCS-A /eth-storage/vlan* # set sharing primary
UCS-A /eth-storage/vlan* # commit-buffer
UCS-A /eth-storage/vlan #

```

Creating a Secondary VLAN for a Private VLAN on an Appliance Cloud



Important

VLANs with IDs from 4030 to 4047 and from 4094 to 4095 are reserved. You cannot create VLANs with IDs from this range. Until Cisco UCS Manager Release 4.0(1d), VLAN ID 4093 was in the list of reserved VLANs. VLAN 4093 has been removed from the list of reserved VLANs and is available for configuration.

The VLAN IDs you specify must also be supported on the switch that you are using. For example, on Cisco Nexus 5000 Series switches, the VLAN ID range from 3968 to 4029 is reserved. Before you specify the VLAN IDs in Cisco UCS Manager, make sure that the same VLAN IDs are available on your switch.

VLANs in the LAN cloud and FCoE VLANs in the SAN cloud must have different IDs. Using the same ID for a VLAN and an FCoE VLAN in a VSAN results in a critical fault and traffic disruption for all vNICs and uplink ports using that VLAN. Ethernet traffic is dropped on any VLAN which has an ID that overlaps with an FCoE VLAN ID.

SUMMARY STEPS

1. UCS-A# **scope eth-storage**
2. UCS-A /eth-storage # **create vlan** *vlan-name* *vlan-id*
3. UCS-A /eth-storage/vlan* # **set sharing isolated**
4. UCS-A /eth-storage/vlan* # **set pubnwnname** *primary-vlan-name*
5. UCS-A /eth-storage/vlan* # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-storage	Enters Ethernet storage mode.
Step 2	UCS-A /eth-storage # create vlan <i>vlan-name</i> <i>vlan-id</i>	Creates a named VLAN, specifies the VLAN name and VLAN ID, and enters Ethernet storage VLAN mode. The VLAN name is case sensitive.
Step 3	UCS-A /eth-storage/vlan* # set sharing isolated	Sets the VLAN as the secondary VLAN.
Step 4	UCS-A /eth-storage/vlan* # set pubnwnname <i>primary-vlan-name</i>	Specifies the primary VLAN to be associated with this secondary VLAN.
Step 5	UCS-A /eth-storage/vlan* # commit-buffer	Commits the transaction to the system configuration.

Example

The following example creates a named VLAN for fabric interconnect A, names the VLAN, assigns the VLAN ID, makes this VLAN the secondary VLAN, associates the secondary VLAN with the primary VLAN, and commits the transaction:

```
UCS-A# scope eth-storage
UCS-A /eth-storage # create vlan isovlan501 501
UCS-A /eth-storage/vlan* # set sharing isolated
UCS-A /eth-storage/vlan* # set pubnwnname primaryvlan500
UCS-A /eth-storage/vlan* # commit-buffer
UCS-A /eth-storage/vlan # #
```

Community VLANs

Cisco UCS Manager supports Community VLANs in UCS Fabric Interconnects. Community ports communicate with each other and with promiscuous ports. Community ports have Layer 2 isolation from all other ports in other communities, or isolated ports within the PVLAN. Broadcasts are transmitted between the community ports associated with the PVLAN only and the other promiscuous ports. A promiscuous port can communicate with all interfaces, including the isolated and community ports within a PVLAN.

Creating a Community VLAN

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**.
2. UCS-A# /eth-uplink/ # **create vlan ID** .
3. UCS-A# /eth-uplink/ vlan # **set sharing Type** .
4. UCS-A# /eth-uplink/ vlan # **set pubnwnname Name** .
5. UCS-A# /eth-uplink/ vlan # **commit-buffer**.

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink .	Enters Ethernet uplink mode.
Step 2	UCS-A# /eth-uplink/ # create vlan ID .	Create a VLAN with the specified VLAN ID.
Step 3	UCS-A# /eth-uplink/ vlan # set sharing Type .	Specifies the vlan type.
Step 4	UCS-A# /eth-uplink/ vlan # set pubnwnname Name .	Specifies the primary vlan association.
Step 5	UCS-A# /eth-uplink/ vlan # commit-buffer .	Commits the transaction to the system configuration.

Example

The following example shows how to create a Community VLAN:

```

UCS-A# scope eth-uplink
UCS-A /eth-uplink # create vlan vlan203 203
UCS-A /eth-uplink/vlan* # set sharing community
UCS-A /eth-uplink/vlan* # set pubname vlan200
UCS-A /eth-uplink/vlan* # commit-buffer
UCS-A /eth-uplink/vlan* # exit
UCS-A /vlan-group #

```

Viewing Community VLANs

SUMMARY STEPS

1. UCS-A# **scope org**
2. UCS-A /org # **show vlan**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope org	Enters Cisco UCS Manager organization.
Step 2	UCS-A /org # show vlan	Displays the available groups in the organization.

Example

The following example shows the available VLAN groups in the root org:

```

UCS-A# scope org
UCS-A# /org/# show vlan
VLAN Group:

```

Name	VLAN ID	Fabric ID	Native VLAN	Sharing Type	Primary Vlan
-----	-----	-----	-----	-----	-----
vlan100	100	Dual	No	Primary	vlan100
vlan100	101	Dual	No	Isolated	vlan100
vlan100	203	Dual	No	Community	vlan200

Allowing Community VLANs on vNICs

SUMMARY STEPS

1. UCS-A# **scope org org-name**
2. UCS-A /org # **scope service-profile profile-name**
3. UCS-A /org/service-profile # **scope vnic vnic-name**
4. UCS-A /org/service-profile/vnic # **create eth-if community-vlan-name**
5. UCS-A /org/service-profile/vnic # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope org <i>org-name</i>	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the <i>org-name</i> .
Step 2	UCS-A /org # scope service-profile <i>profile-name</i>	Commits the transaction to the system configuration.
Step 3	UCS-A /org/service-profile # scope vnic <i>vnic-name</i>	Enters command mode for the specified vNIC.
Step 4	UCS-A /org/service-profile/vnic # create eth-if <i>community-vlan-name</i>	Allows the community VLAN to access the specified vNIC.
Step 5	UCS-A /org/service-profile/vnic # commit-buffer	Commits the transaction to the system configuration.

Example

The following example shows how to assign the community VLAN cVLAN101 to the vNIC vnic_1 and commits the transaction.

```
UCS-A# scope org /
UCS-A /org # scope service-profile GSP1
UCS-A /org/service-profile # scope vnic vnic_1
UCS-A /org/service-profile/vnic # create eth-if cVLAN101
UCS-A /org/service-profile/vnic* # commit-buffer
```

Allowing PVLAN on Promiscuous Access or Trunk Port

For a promiscuous access port, the isolated and community VLANs must be associated to the same primary VLAN.

For a promiscuous trunk port, isolated and community VLANs belonging to different primary VLANs are allowed, as well as regular VLANs.

SUMMARY STEPS

1. UCS-A # **scope eth-storage**
2. UCS-A /eth-storage # **scope vlan** *iso-vlan-name*
3. UCS-A /eth-storage/vlan # **create member-port** *fabric slot- num port- num*
4. UCS-A /eth-storage/vlan/member-port # **exit**
5. UCS-A /eth-storage/vlan # **exit**
6. UCS-A /eth-storage # **scope vlan** *comm-vlan-name*
7. UCS-A /eth-storage/vlan # **create member-port** *fabric slot- num port- num*
8. UCS-A /eth-storage/vlan/member-port # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A # scope eth-storage	Enters Ethernet storage mode.

	Command or Action	Purpose
Step 2	UCS-A /eth-storage # scope vlan <i>iso-vlan-name</i>	Enters the specified isolated VLAN.
Step 3	UCS-A /eth-storage/vlan # create member-port <i>fabric slot- num port- num</i>	Creates the member port for the specified fabric, assigns the slot number and port number, and enters member port configuration scope.
Step 4	UCS-A /eth-storage/vlan/member-port # exit	Returns to VLAN mode.
Step 5	UCS-A /eth-storage/vlan # exit	Returns to Ethernet storage mode.
Step 6	UCS-A /eth-storage # scope vlan <i>comm-vlan-name</i>	Enters the specified community VLAN.
Step 7	UCS-A /eth-storage/vlan # create member-port <i>fabric slot- num port- num</i>	Creates the member port for the specified fabric, assigns the slot number and port number, and enters member port configuration scope.
Step 8	UCS-A /eth-storage/vlan/member-port # commit-buffer	Commits the transaction to the system configuration.

Example

The following example shows how to assign the isolated and community associated with the same primary VLAN to the same appliance port and commits the transaction.

```
UCS-A# scope eth-storage
UCS-A /eth-storage # scope vlan isovlan501
UCS-A /eth-storage/vlan # create member-port a 1 2
UCS-A /eth-storage/vlan/member-port* # exit
UCS-A /eth-storage/vlan* # exit
UCS-A /eth-storage* # scope vlan cvlan502
UCS-A /eth-storage/vlan* # create member-port a 1 2
UCS-A /eth-storage/vlan/member-port* # commit-buffer
UCS-A /eth-storage/vlan/member-port #
```

Deleting a Community VLAN

If Cisco UCS Manager includes a named VLAN with the same VLAN ID as the one you delete, the VLAN is not removed from the fabric interconnect configuration until all named VLANs with that ID are deleted.

If you are deleting a private primary VLAN, ensure that you reassign the secondary VLANs to another working primary VLAN.

Before you begin

Before you delete a VLAN from a fabric interconnect, ensure that the VLAN was removed from all vNICs and vNIC templates.



Note

If you delete a VLAN that is assigned to a vNIC or vNIC template, the vNIC might allow that VLAN to flap.

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. (Optional) UCS-A /eth-uplink # **scope fabric {a | b}**
3. UCS-A /eth-uplink # **delete community vlan vlan-name**
4. UCS-A /eth-uplink # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	(Optional) UCS-A /eth-uplink # scope fabric {a b}	Enters Ethernet uplink fabric mode. Use this command when you want to delete a named VLAN only from the specified fabric (a or b).
Step 3	UCS-A /eth-uplink # delete community vlan vlan-name	Deletes the specified community VLAN.
Step 4	UCS-A /eth-uplink # commit-buffer	Commits the transaction to the system configuration.

Example

The following example deletes a Community VLAN and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # delete community vlan vlan203
UCS-A /eth-uplink* # commit-buffer
UCS-A /eth-uplink #
```

Viewing the VLAN Port Count

SUMMARY STEPS

1. UCS-A# **scope fabric-interconnect {a | b}**
2. UCS-A /fabric-interconnect # **show vlan-port-count**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope fabric-interconnect {a b}	Enters fabric interconnect mode for the specified fabric interconnect.
Step 2	UCS-A /fabric-interconnect # show vlan-port-count	Displays the VLAN port count.

Example

The following example displays the VLAN port count for fabric interconnect A:

```
UCS-A# scope fabric-interconnect a
UCS-A /fabric-interconnect # show vlan-port-count
```

```
VLAN-Port Count:
VLAN-Port Limit      Access VLAN-Port Count      Border VLAN-Port Count      Alloc Status
-----
6000                  3                               0                            Available
```

VLAN Port Count Optimization

VLAN port count optimization enables mapping the state of multiple VLANs into a single internal state. When you enable the VLAN port count optimization, Cisco UCS Manager logically groups VLANs based on the port VLAN membership. This grouping increases the port VLAN count limit. VLAN port count optimization also compresses the VLAN state and reduces the CPU load on the fabric interconnect. This reduction in the CPU load enables you to deploy more VLANs over more vNICs. Optimizing VLAN port count does not change any of the existing VLAN configuration on the vNICs.

VLAN port count optimization is disabled by default. You can enable or disable the option based on your requirements.



Important

- Enabling VLAN port count optimization increases the number of available VLAN ports for use. If the port VLAN count exceeds the maximum number of VLANs in a non-optimized state, you cannot disable the VLAN port count optimization.
- VLAN port count optimization is not supported in Cisco UCS 6100 Series fabric interconnect.

On the Cisco UCS 6400 Series Fabric Interconnect, VLAN port count optimization is performed when the PV count exceeds 16000.

When the Cisco UCS 6400 Series Fabric Interconnect is in Ethernet switching mode:

- The FI does not support **VLAN Port Count Optimization Enabled**
- The FI supports 16000 PVs, similar to EHM mode, when **VLAN Port Count Optimization is Disabled**

The following table illustrates the PV Count with VLAN port count optimization enabled and disabled on UCS 6200, 6300, and Cisco UCS 6400 Series Fabric Interconnects.

	6200 Series FI	6300 Series FI	6400 Series FI
PV Count with VLAN Port Count Optimization Disabled	32000	16000	16000
PV Count with VLAN Port Count Optimization Enabled	64000	64000	64000

Enabling Port VLAN Count Optimization

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink# **set vlan-port-count-optimization enable**
3. UCS-A /eth-uplink* # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink# set vlan-port-count-optimization enable	Enables the vlan for port VLAN count optimization.
Step 3	UCS-A /eth-uplink* # commit-buffer	Commits the transaction to the system configuration.

Example

The following example shows how to enable VLAN port count optimization:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # set vlan-port-count-optimization enable
UCS-A /eth-uplink* # commit-buffer
UCS-A /eth-uplink#
```

Disabling Port VLAN Count Optimization

If you have more Port VLAN count than that is allowed in the non port VLAN port count optimization state, you cannot disable the optimization.

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink# **set vlan-port-count-optimization disable**
3. UCS-A /eth-uplink # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink# set vlan-port-count-optimization disable	Disables the port VLAN count optimization.
Step 3	UCS-A /eth-uplink # commit-buffer	Commits the transaction to the system configuration.

Example

The following example shows how to disable VLAN port count optimization:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # set vlan-port-count-optimization disable
UCS-A /eth-uplink* # commit-buffer
UCS-A /eth-uplink#
```

Viewing the Port VLAN Count Optimization Groups

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink# **show vlan-port-count-optimization group**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink# show vlan-port-count-optimization group	Displays the vlan for port VLAN count optimization groups.

Example

The following example shows port VLAN count optimization group in fabric a and b:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # show vlan-port-count-optimization group
VLAN Port Count Optimization Group:
  Fabric ID  Group ID  VLAN ID
  -----
  A          5         6
  A          5         7
  A          5         8
  B          10        100
  B          10        101
```

VLAN Groups

VLAN groups allow you to group VLANs on Ethernet uplink ports, by function or by VLANs that belong to a specific network. You can define VLAN membership and apply the membership to multiple Ethernet uplink ports on the fabric interconnect.

**Note**

Cisco UCS Manager supports a maximum of 200 VLAN Groups. If Cisco UCS Manager determines that you create more than 200 VLAN groups, the system disables VLAN compression.

You can configure inband and out-of-band (OOB) VLAN groups to use to access the Cisco Integrated Management Interface (CIMC) on blade and rack servers. Cisco UCS Manager supports OOB IPv4 and inband IPv4 and IPv6 VLAN groups for use with the uplink interfaces or uplink port channels.



Note Inband Management is not supported on VLAN 2 or VLAN 3.

After you assign a VLAN to a VLAN group, any changes to the VLAN group are applied to all Ethernet uplink ports that are configured with the VLAN group. The VLAN group also enables you to identify VLAN overlaps between disjoint VLANs.

You can configure uplink ports under a VLAN group. When you configure an uplink port for a VLAN group, that uplink port will support all the VLANs that are part of the associated VLAN groups and individual VLANs that are associated with the uplink using LAN Uplinks Manager, if any. Further, any uplink that is not selected for association with that VLAN group will stop supporting the VLANs that are part of that VLAN group.

You can create VLAN groups from the **LAN Cloud** or from the **LAN Uplinks Manager**.

Creating a VLAN Group

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**.
2. UCS-A# /eth-uplink/ **#create vlan-group***Name* .
3. UCS-A# /eth-uplink/ vlan-group**#create member-vlan***ID* .
4. UCS-A# /eth-uplink/vlan-group **#create member-port** [member-port-channel] .
5. UCS-A#/vlan-group* **# commit-buffer**.

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink .	Enters Ethernet uplink mode. The VLAN Group name is case sensitive.
Step 2	UCS-A# /eth-uplink/ #create vlan-group <i>Name</i> .	Create a VLAN group with the specified name. This name can be between 1 and 32 alphanumeric characters. You cannot use spaces or any special characters other than - (hyphen), _ (underscore), : (colon), and . (period), and you cannot change this name after the object is saved.
Step 3	UCS-A# /eth-uplink/ vlan-group #create member-vlan <i>ID</i> .	Adds the specified VLANs to the created VLAN group.
Step 4	UCS-A# /eth-uplink/vlan-group #create member-port [member-port-channel] .	Assigns the uplink Ethernet ports to the VLAN group.
Step 5	UCS-A#/vlan-group* # commit-buffer .	Commits the transaction to the system configuration.

Example

The following example shows how to create a VLAN group:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # create vlan-group eng
UCS-A /eth-uplink/vlan-group* # create member-vlan 3
UCS-A /eth-uplink/vlan-group* # commit-buffer
UCS-A /vlan-group #
```

Creating an Inband VLAN Group

Configure inband VLAN groups to provide access to remote users via an inband service profile.

SUMMARY STEPS

1. UCS-A# **scope eth uplink**
2. UCS-A /eth-uplink # **create vlan-group inband-vlan-name**
3. UCS-A /eth-uplink/vlan-group # **create member-vlan inband-vlan-name inband-vlan-id**
4. UCS-A /eth-uplink/vlan-group/member-vlan # **exit**
5. UCS-A /eth-uplink/vlan-group # **create member-port fabricslot-num port-num**
6. UCS-A /eth-uplink/vlan-group/member-port # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth uplink	Enters Ethernet uplink configuration mode.
Step 2	UCS-A /eth-uplink # create vlan-group inband-vlan-name	Creates a VLAN group with the specified name and enters VLAN group configuration mode.
Step 3	UCS-A /eth-uplink/vlan-group # create member-vlan inband-vlan-name inband-vlan-id	Adds the specified VLAN to the VLAN group and enters VLAN group member configuration mode.
Step 4	UCS-A /eth-uplink/vlan-group/member-vlan # exit	Exits VLAN group member configuration mode.
Step 5	UCS-A /eth-uplink/vlan-group # create member-port fabricslot-num port-num	Creates the member port for the specified fabric, assigns the slot number, and port number and enters member port configuration.
Step 6	UCS-A /eth-uplink/vlan-group/member-port # commit-buffer	Commits the transaction.

Example

The example below creates a VLAN group named inband-vlan-group, creates a member of the group named Inband_VLAN and assigns VLAN ID 888, creates member ports for Fabric A and Fabric B, and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # create vlan-group inband-vlan-group
```

```

UCS-A /eth-uplink/vlan-group* # create member-vlan Inband_VLAN 888
UCS-A /eth-uplink/vlan-group/member-vlan* # exit
UCS-A /eth-uplink/vlan-group* # create member-port a 1 23
UCS-A /eth-uplink/vlan-group/member-port* # exit
UCS-A /eth-uplink/vlan-group* # create member-port b 1 23
UCS-A /eth-uplink/vlan-group/member-port* # commit-buffer
UCS-A /eth-uplink/vlan-group/member-port # exit
UCS-A /eth-uplink/vlan-group # exit

```

What to do next

Assign the inband VLAN group to an inband service profile.

Viewing VLAN Groups

SUMMARY STEPS

1. UCS-A# **scope org**
2. UCS-A /org # **show vlan-group**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope org	Enters Cisco UCS Manager organization.
Step 2	UCS-A /org # show vlan-group	Displays the available groups in the organization.

Example

The following example shows the available VLAN groups in the root org:

```

UCS-A# scope org
UCS-A# /org/# show vlan-group
VLAN Group:
  Name
  ----
  eng
  hr
  finance

```

Deleting a VLAN Group

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**.
2. UCS-A# /eth-uplink/ **#delete vlan-groupName** .
3. UCS-A#/eth-uplink* **# commit-buffer**.

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink.	Enters Ethernet uplink mode.
Step 2	UCS-A# /eth-uplink/ #delete vlan-group <i>Name</i> .	Deletes the specified VLAN group.
Step 3	UCS-A#/eth-uplink* # commit-buffer.	Commits the transaction to the system configuration.

Example

The following example shows how to delete a VLAN group:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # delete vlan-group eng
UCS-A /eth-uplink* # commit-buffer
UCS-A /eth-uplink #
```

Modifying the Reserved VLAN

This task describes how to modify the reserved VLAN ID. Modifying the reserved VLAN makes transitioning from Cisco UCS 6200 Series Fabric Interconnects to the Cisco UCS 6454 Fabric Interconnect more flexible with preexisting network configurations. The reserved VLAN block is configurable by assigning a contiguous block of 128 unused VLANs, rather than reconfiguring the currently existing VLANs that conflict with the default range. For example, if the reserved VLAN is changed to 3912, then the new VLAN block range spans 3912 to 4039. You can select any contiguous block of 128 VLAN IDs, with the start ID ranging from 2 to 3915. Changing the reserved VLAN requires a reload of the 6454 Fabric Interconnect for the new values to take effect.

SUMMARY STEPS

1. UCS-A# **scope eth-uplink.**
2. UCS-A# /eth-uplink/ **#show reserved-vlan** .
3. UCS-A# /eth-uplink/ **#scope reserved-vlan.**
4. UCS-A# /eth-uplink/reserved-vlan **#set start-vlan-id** [vlan-id] .
5. UCS-A# /eth-uplink/reserved-vlan* **# commit-buffer.**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink.	Enters Ethernet uplink mode.
Step 2	UCS-A# /eth-uplink/ #show reserved-vlan .	This displays the reserved VLAN IDs.
Step 3	UCS-A# /eth-uplink/ #scope reserved-vlan.	Enters reserved VLAN ID specification mode.
Step 4	UCS-A# /eth-uplink/reserved-vlan #set start-vlan-id [vlan-id] .	Assigns the new reserved VLAN starting ID. The reserved VLAN range ID can be specified from 2-3915.
Step 5	UCS-A# /eth-uplink/reserved-vlan* # commit-buffer.	Commits the transaction to the system configuration.

Example

The following example shows how to modify the reserved VLAN ID:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # show reserved-vlan
UCS-A /eth-uplink/ # scope reserved-vlan
UCS-A /eth-uplink/reserved-vlan # set start-vlan-id 3912
UCS-A /eth-uplink/reserved-vlan/* # commit-buffer
```

VLAN Permissions

VLAN permissions restrict access to VLANs based on specified organizations and on the service profile organizations to which the VLANs belong. VLAN permissions also restrict the set of VLANs that you can assign to service profile vNICs. VLAN permissions is an optional feature and is disabled by default. You can enable or disable the feature based on your requirements. If you disable the feature, all of the VLANs are globally accessible to all organizations.



Note If you enable the org permission in **LAN > LAN Cloud > Global Policies > Org Permissions**, when you create a VLAN, the **Permitted Orgs for VLAN(s)** option displays in the **Create VLANs** dialog box. If you do not enable the **Org Permissions**, the **Permitted Orgs for VLAN(s)** option does not display.

Enabling the org permission allows you to specify the organizations for the VLAN. When you specify the organizations, the VLAN becomes available to that specific organization and all of the sub organizations below the structure. Users from other organizations cannot access this VLAN. You can also modify the VLAN permission anytime based on changes to your VLAN access requirements.



Caution When you assign the VLAN org permission to an organization at the root level, all sub organizations can access the VLANs. After assigning the org permission at the root level, and you change the permission for a VLAN that belongs to a sub organization, that VLAN becomes unavailable to the root level organization.

Creating VLAN Permissions

SUMMARY STEPS

1. UCS-A# **scope org**.
2. UCS-A# /org/ #**create vlan-permit***VLAN permission name*.
3. UCS-A#/org* # **commit-buffer**.

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope org .	Enters the Cisco UCS Manager VLAN organization.

	Command or Action	Purpose
Step 2	UCS-A# /org/ # create vlan-permit <i>/VLAN permission name.</i>	Creates the specified VLAN permission and assigns VLAN access permission to the organization.
Step 3	UCS-A# /org* # commit-buffer.	Commits the transaction to the system configuration.

Example

The following example shows how to create a VLAN permission for an organization:

```
UCS-A# scope org
UCS-A /org # create vlan-permit dev
UCS-A /org* # commit-buffer
UCS-A /org #
```

Viewing VLAN Permissions

SUMMARY STEPS

1. UCS-A# **scope org**
2. UCS-A /org # **show vlan-permit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope org	Enters Cisco UCS Manager organization.
Step 2	UCS-A /org # show vlan-permit	Displays the available permissions in the organization.

Example

The following example shows the VLAN groups that have permission to access this VLAN:

```
UCS-A# scope org
UCS-A# /org/# show vlan-permit
VLAN Group:
  Name
  ----
  eng
  hr
  finance
```

Deleting a VLAN Permission

SUMMARY STEPS

1. UCS-A# **scope org.**

2. UCS-A# /org/ #**delete vlan-permit***VLAN permission name*.
3. UCS-A#/org* # **commit-buffer**.

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope org .	Enters the Cisco UCS Manager VLAN organization.
Step 2	UCS-A# /org/ # delete vlan-permit <i>VLAN permission name</i> .	Deletes the access permission to the VLAN.
Step 3	UCS-A#/org* # commit-buffer .	Commits the transaction to the system configuration.

Example

The following example shows how to delete a VLAN permission from an organization:

```
UCS-A# scope org
UCS-A /org # delete vlan-permit dev
UCS-A /org* # commit-buffer
UCS-A /org #
```




CHAPTER 6

LAN Pin Groups

- [LAN Pin Groups, on page 141](#)
- [Configuring a LAN Pin Group, on page 141](#)

LAN Pin Groups

Cisco UCS uses LAN pin groups to pin Ethernet traffic from a vNIC on a server to an uplink Ethernet port or port channel on the fabric interconnect. You can use this pinning to manage the distribution of traffic from the servers.

To configure pinning for a server, you must include the LAN pin group in a vNIC policy. The vNIC policy is then included in the service profile assigned to that server. All traffic from the vNIC travels through the I/O module to the specified uplink Ethernet port.



Note

If you do not assign a pin group to a server interface through a vNIC policy, Cisco UCS Manager chooses an uplink Ethernet port or port channel for traffic from that server interface dynamically. This choice is not permanent. A different uplink Ethernet port or port channel may be used for traffic from that server interface after an interface flap or a server reboot.

If an uplink is part of a LAN pin group, the uplink is not necessarily reserved for only that LAN pin group. Other vNIC's policies that do not specify a LAN pin group can use the uplink as a dynamic uplink.

Configuring a LAN Pin Group

In a system with two fabric interconnects, you can associate the pin group with only one fabric interconnect or with both fabric interconnects.

Before you begin

Configure the ports and port channels with which you want to configure the pin group. You can only include ports and port channels configured as uplink ports in a LAN pin group.

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**

2. UCS-A /eth-uplink # **create pin-group** *pin-group-name*
3. (Optional) UCS-A /eth-uplink/pin-group # **set descr** *description*
4. (Optional) UCS-A /eth-uplink/pin-group # **set target** {a | b | dual} {port slot-num / port-num | port-channel port-num}
5. UCS-A /eth-uplink/pin-group # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # create pin-group <i>pin-group-name</i>	Creates an Ethernet (LAN) pin group with the specified name, and enters Ethernet uplink pin group mode.
Step 3	(Optional) UCS-A /eth-uplink/pin-group # set descr <i>description</i>	Provides a description for the pin group. Note If your description includes spaces, special characters, or punctuation, you must begin and end your description with quotation marks. The quotation marks will not appear in the description field of any show command output.
Step 4	(Optional) UCS-A /eth-uplink/pin-group # set target {a b dual} {port slot-num / port-num port-channel port-num}	Sets the Ethernet pin target to the specified fabric and port, or fabric and port channel.
Step 5	UCS-A /eth-uplink/pin-group # commit-buffer	Commits the transaction to the system configuration.

Example

The following example creates a LAN pin group named pingroup54 on fabric A, provides a description for the pin group, sets the pin group target to port channel 28, and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # create pin-group pingroup54
UCS-A /eth-uplink/pin-group* # set descr "This is my pin group #54"
UCS-A /eth-uplink/pin-group* # set target a port-channel 28
UCS-A /eth-uplink/pin-group* # commit-buffer
UCS-A /eth-uplink/pin-group #
```

What to do next

Include the pin group in a vNIC template.



CHAPTER 7

MAC Pools

- [MAC Pools, on page 143](#)
- [Creating a MAC Pool, on page 143](#)
- [Deleting a MAC Pool, on page 145](#)

MAC Pools

A MAC pool is a collection of network identities, or MAC addresses, that are unique in their Layer 2 environment and are available to be assigned to vNICs on a server. If you use MAC pools in service profiles, you do not have to manually configure the MAC addresses to be used by the server associated with the service profile.

In a system that implements multitenancy, you can use the organizational hierarchy to ensure that MAC pools can be used only by specific applications or business services. Cisco UCS uses the name resolution policy to assign MAC addresses from the pool.

To assign a MAC address to a server, you must include the MAC pool in a vNIC policy. The vNIC policy is then included in the service profile assigned to that server.

You can specify your own MAC addresses or use a group of MAC addresses provided by Cisco.

Creating a MAC Pool

SUMMARY STEPS

1. UCS-A# **scope org** *org-name*
2. UCS-A /org # **create mac-pool** *mac-pool-name*
3. (Optional) UCS-A /org/mac-pool # **set descr** *description*
4. UCS-A /org/mac-pool # **set assignmentorder** {**default** | **sequential**}
5. UCS-A /org/mac-pool # **create block** *first-mac-addr* *last-mac-addr*
6. UCS-A /org/mac-pool # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope org <i>org-name</i>	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the <i>org-name</i> .
Step 2	UCS-A /org # create mac-pool <i>mac-pool-name</i>	Creates a MAC pool with the specified name, and enters organization MAC pool mode. This name can be between 1 and 32 alphanumeric characters. You cannot use spaces or any special characters other than - (hyphen), _ (underscore), : (colon), and . (period), and you cannot change this name after the object is saved.
Step 3	(Optional) UCS-A /org/mac-pool # set descr <i>description</i>	Provides a description for the MAC pool. Note If your description includes spaces, special characters, or punctuation, you must begin and end your description with quotation marks. The quotation marks will not appear in the description field of any show command output.
Step 4	UCS-A /org/mac-pool # set assignmentorder { default sequential }	This can be one of the following: <ul style="list-style-type: none"> • default—Cisco UCS Manager selects a random identity from the pool. • sequential—Cisco UCS Manager selects the lowest available identity from the pool.
Step 5	UCS-A /org/mac-pool # create block <i>first-mac-addr last-mac-addr</i>	Creates a block (range) of MAC addresses, and enters organization MAC pool block mode. You must specify the first and last MAC addresses in the address range using the form <i>nn:nn:nn:nn:nn:nn</i> , with the addresses separated by a space. Note A MAC pool can contain more than one MAC address block. To create multiple MAC address blocks, you must enter multiple create block commands from organization MAC pool mode.
Step 6	UCS-A /org/mac-pool # commit-buffer	Commits the transaction to the system configuration.

Example

The following example shows how to create a MAC pool named pool37, provide a description for the pool, define a MAC address block by specifying the first and last MAC addresses in the block, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org # create mac-pool pool37
UCS-A /org/mac-pool* # set descr "This is my MAC pool"
```

```
UCS-A /org/mac-pool* # create block 00:A0:D7:42:00:01 00:A0:D7:42:01:00
UCS-A /org/mac-pool/block* # commit-buffer
UCS-A /org/mac-pool/block #
```

What to do next

Include the MAC pool in a vNIC template.

Deleting a MAC Pool

If you delete a pool, Cisco UCS Manager does not reallocate any addresses from that pool that were assigned to vNICs or vHBAs. All assigned addresses from a deleted pool remain with the vNIC or vHBA to which they are assigned until one of the following occurs:

- The associated service profiles are deleted.
- The vNIC or vHBA to which the address is assigned is deleted.
- The vNIC or vHBA is assigned to a different pool.

SUMMARY STEPS

1. UCS-A# **scope org** *org-name*
2. UCS-A /org # **delete mac-pool** *pool-name*
3. UCS-A /org # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope org <i>org-name</i>	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the <i>org-name</i> .
Step 2	UCS-A /org # delete mac-pool <i>pool-name</i>	Deletes the specified MAC pool.
Step 3	UCS-A /org # commit-buffer	Commits the transaction to the system configuration.

Example

The following example shows how to delete the MAC pool named pool4 and commit the transaction:

```
UCS-A# scope org /
UCS-A /org # delete mac-pool pool4
UCS-A /org* # commit-buffer
UCS-A /org #
```




CHAPTER 8

Quality of Service

- [Quality of Service, on page 147](#)
- [Configuring System Classes, on page 148](#)
- [Configuring Quality of Service Policies, on page 152](#)
- [Configuring Flow Control Policies, on page 155](#)
- [Configuring Slow Drain, on page 157](#)

Quality of Service

Cisco UCS provides the following methods to implement quality of service:

- System classes that specify the global configuration for certain types of traffic across the entire system
- QoS policies that assign system classes for individual vNICs
- Flow control policies that determine how uplink Ethernet ports handle pause frames

Global QoS changes made to the QoS system class may result in brief data-plane interruptions for all traffic. Some examples of such changes are:

- Changing the MTU size for an enabled class
- Changing packet drop for an enabled class
- Changing the CoS value for an enabled class

Guidelines and Limitations for Quality of Service on Cisco UCS 6300 Series Fabric Interconnect

- Cisco UCS 6300 Series Fabric Interconnect uses a shared buffer for all system classes.
- Multicast optimization is not supported.
- When you change the QoS parameters for any class causes traffic disruption to all classes. The following table lists the changes in the QoS system class and the conditions that trigger a system reboot.

QoS System class status	Condition	FI Reboot Status
Enabled	Change between drop and no drop	Yes

QoS System class status	Condition	FI Reboot Status
No-drop	Change between enable and disable	Yes
Enable and no-drop	Change in MTU size	Yes

- The subordinate FI reboots first as a result of the change in the QoS system class, followed by the primary FI.

**Note**

Cisco UCS Manager displays a prompt that the fabric interconnect will reboot when the system policy is changed.

- **show queuing interface** command is not supported.

Guidelines and Limitations for Quality of Service on Cisco UCS Mini

- Cisco UCS Mini uses a shared buffer for all system classes.
- The bronze class shares the buffer with SPAN. We recommend using either SPAN or the bronze class.
- Multicast optimization is not supported.
- Changing the QoS parameters for any class causes traffic disruption to all classes.
- When mixing Ethernet and FC or FCoE traffic, the bandwidth distribution is not equal.
- Multiple streams of traffic from the same class may not be distributed equally.
- Use the same CoS values for all no-drop policies to avoid any FC or FCoE performance issues.
- Only the platinum and gold classes support no-drop policies.
- **show queuing interface** command is not supported.

Configuring System Classes

System Classes

Cisco UCS uses Data Center Ethernet (DCE) to handle all traffic inside a Cisco UCS domain. This industry standard enhancement to Ethernet divides the bandwidth of the Ethernet pipe into eight virtual lanes. Two virtual lanes are reserved for internal system and management traffic. You can configure quality of service (QoS) for the other six virtual lanes. System classes determine how the DCE bandwidth in these six virtual lanes is allocated across the entire Cisco UCS domain.

Each system class reserves a specific segment of the bandwidth for a specific type of traffic, which provides a level of traffic management, even in an oversubscribed system. For example, you can configure the **Fibre Channel Priority** system class to determine the percentage of DCE bandwidth allocated to FCoE traffic.

The following table describes the system classes that you can configure.

Table 5: System Classes

System Class	Description
Platinum Gold Silver Bronze	<p>A configurable set of system classes that you can include in the QoS policy for a service profile. Each system class manages one lane of traffic.</p> <p>All properties of these system classes are available for you to assign custom settings and policies.</p> <p>For Cisco UCS Mini, packet drop can only be disabled on the platinum and gold classes. Only one platinum and one gold class can be configured as a no drop class at a time.</p>
Best Effort	<p>A system class that sets the quality of service for the lane reserved for basic Ethernet traffic.</p> <p>Some properties of this system class are preset and cannot be modified. For example, this class has a drop policy that allows it to drop data packets if required. You cannot disable this system class.</p>
Fibre Channel	<p>A system class that sets the quality of service for the lane reserved for Fibre Channel over Ethernet traffic.</p> <p>Some properties of this system class are preset and cannot be modified. For example, this class has a no-drop policy that ensures it never drops data packets. You cannot disable this system class.</p> <p>Note FCoE traffic has a reserved QoS system class that should not be used by any other type of traffic. If any other type of traffic has a CoS value that is used by FCoE, the value is remarked to 0.</p>

Configuring a System Class

The type of adapter in a server might limit the maximum MTU supported. For example, network MTU above the maximums might cause the packet to be dropped for the following adapters:

- The Cisco UCS M71KR CNA adapter, which supports a maximum MTU of 9216.
- The Cisco UCS 82598KR-CI adapter, which supports a maximum MTU of 14000.

SUMMARY STEPS

1. UCS-A# **scope eth-server**
2. UCS-A /eth-server # **scope qos**
3. UCS-A /eth-server/qos # **scope eth-classified {bronze | gold | platinum | silver}**
4. UCS-A /eth-server/qos/eth-classified # **enable**
5. UCS-A /eth-server/qos/eth-classified # **set cos** *cos-value*
6. UCS-A /eth-server/qos/eth-classified # **set drop** {drop | no-drop}
7. UCS-A /eth-server/qos/eth-classified # **set mtu** {mtu-value | fc | normal}
8. UCS-A /eth-server/qos/eth-classified # **set multicast-optimize** {no | yes}
9. UCS-A /eth-server/qos/eth-classified # **set weight** {weight-value | best-effort | none}

10. UCS-A /eth-server/qos/eth-classified # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-server	Enters Ethernet server mode.
Step 2	UCS-A /eth-server # scope qos	Enters Ethernet server QoS mode.
Step 3	UCS-A /eth-server/qos # scope eth-classified {bronze gold platinum silver}	Enters Ethernet server QoS Ethernet classified mode for the specified system class.
Step 4	UCS-A /eth-server/qos/eth-classified # enable	Enables the specified system class.
Step 5	UCS-A /eth-server/qos/eth-classified # set cos <i>cos-value</i>	<p>Specifies the class of service for the specified system class. Valid class of service values are 0 to 6.</p> <p>Important Use the same CoS values on UCS and N5K for all the no-drop policies. To insure that end-to-end PFC works correctly, have the same QoS policy configured on all intermediate switches.</p> <p>Note When the CoS value is set to 0 in any QoS class, this causes the adapter to use the same queue for best effort and the QoS class. When traffic congestion occurs, best effort and the QoS class will share the bandwidth equally instead of using the weight configured in the QoS class.</p>
Step 6	UCS-A /eth-server/qos/eth-classified # set drop {drop no-drop}	<p>Specifies whether the channel can drop packets or not.</p> <p>Note Changes saved to the drop displays the following warning message: Warning: The operation will cause momentary disruption to traffic forwarding.</p>
Step 7	UCS-A /eth-server/qos/eth-classified # set mtu {mtu-value fc normal}	<p>The maximum transmission unit, or packet size to be used. The maximum value for MTU is 9216.</p> <p>Note If the vNIC has an associated QoS policy, the MTU specified here must be equal to or less than the MTU specified in the associated QoS system class. If this MTU value exceeds the MTU value in the QoS system class, packets might get dropped during data transmission.</p> <p>Changes saved to the MTU displays the following warning message: Warning: The operation will cause momentary disruption to traffic forwarding.</p>
Step 8	UCS-A /eth-server/qos/eth-classified # set multicast-optimize {no yes}	Specifies whether the class is optimized to for sending multicast packets.

	Command or Action	Purpose
Step 9	UCS-A /eth-server/qos/eth-classified # set weight { <i>weight-value</i> best-effort none }	Specifies the relative weight for the specified system class. Valid weight values are 0 to 10.
Step 10	UCS-A /eth-server/qos/eth-classified # commit-buffer	Commits the transaction to the system configuration.

Example

The following example shows how to enable the platinum system class, allow the channel to drop packets, set the class of service to 6, set the MTU to normal, set the relative weight to 5, and commit the transaction:

```
UCS-A# scope eth-server
UCS-A /eth-server # scope qos
UCS-A /eth-server/qos # scope eth-classified platinum
UCS-A /eth-server/qos/eth-classified # enable
UCS-A /eth-server/qos/eth-classified* # set drop drop
Warning: The operation will cause momentary disruption to traffic forwarding
UCS-A /eth-server/qos/eth-classified* # set cos 6
UCS-A /eth-server/qos/eth-classified* # set mtu normal
Warning: The operation will cause momentary disruption to traffic forwarding
UCS-A /eth-server/qos/eth-classified* # set weight 5
UCS-A /eth-server/qos/eth-classified* # commit-buffer
UCS-A /eth-server/qos/eth-classified #
```

Disabling a System Class

If you disable a system class that is used in a QoS policy, Cisco UCS Manager uses the system class configured with CoS 0 for traffic on servers that are configured with the QoS policy. If no system class is configured as CoS 0, the Best Effort system class is used. You cannot disable the Best Effort or Fibre Channel system classes.

SUMMARY STEPS

1. UCS-A# **scope eth-server**
2. UCS-A /eth-server # **scope qos**
3. UCS-A /eth-server/qos # **scope eth-classified** {**bronze** | **gold** | **platinum** | **silver**}
4. UCS-A /eth-server/qos/eth-classified # **disable**
5. UCS-A /eth-server/qos/eth-classified # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-server	Enters Ethernet server mode.
Step 2	UCS-A /eth-server # scope qos	Enters Ethernet server QoS mode.
Step 3	UCS-A /eth-server/qos # scope eth-classified { bronze gold platinum silver }	Enters Ethernet server QoS Ethernet classified mode for the specified system class.
Step 4	UCS-A /eth-server/qos/eth-classified # disable	Disables the specified system class.

	Command or Action	Purpose
Step 5	UCS-A /eth-server/qos/eth-classified # commit-buffer	Commits the transaction to the system configuration.

Example

The following example disables the platinum system class and commits the transaction:

```
UCS-A# scope eth-server
UCS-A /eth-server # scope qos
UCS-A /eth-server/qos # scope eth-classified platinum
UCS-A /eth-server/qos/eth-classified # disable
UCS-A /eth-server/qos/eth-classified* # commit-buffer
UCS-A /eth-server/qos/eth-classified #
```

Configuring Quality of Service Policies

Quality of Service Policy

A quality of service (QoS) policy assigns a system class to the outgoing traffic for a vNIC or vHBA. This system class determines the quality of service for that traffic. For certain adapters, you can also specify additional controls on the outgoing traffic, such as burst and rate.

You must include a QoS policy in a vNIC policy or vHBA policy and then include that policy in a service profile to configure the vNIC or vHBA.

Configuring a QoS Policy

SUMMARY STEPS

1. Switch-A# **scope org** *org-name*
2. Switch-A /org # **create qos-policy** *policy-name*
3. Switch-A /org/qos-policy # **create egress-policy**
4. Switch-A /org/qos-policy/egress-policy # **set host-cos-control** {full | none}
5. Switch-A /org/qos-policy/egress-policy # **set prio** *sys-class-name*
6. Switch-A /org/qos-policy/egress-policy # **set rate** {line-rate | kbps} **burst** *bytes*
7. Switch-A /org/qos-policy/egress-policy # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Switch-A# scope org <i>org-name</i>	Enters org mode for the specified organization. To enter the default org mode, type / as the <i>org-name</i> .
Step 2	Switch-A /org # create qos-policy <i>policy-name</i>	Creates the specified QoS policy, and enters org QoS policy mode.

	Command or Action	Purpose
Step 3	Switch-A /org/qos-policy # create egress-policy	Creates the egress policy (for both vNICs and vHBAs) to be used by the QoS policy, and enters org QoS policy egress policy mode.
Step 4	Switch-A /org/qos-policy/egress-policy # set host-cos-control {full none}	<p>(Optional) Specifies whether the host or Cisco UCS Manager controls the class of service (CoS) for a vNIC. This setting has no effect on a vHBA.</p> <p>Use the full keyword to have the host control the CoS. If the packet has a valid CoS value, the host uses that value. Otherwise, it uses the CoS value associated with the specified class priority. Use the none keyword to have Cisco UCS Manager use the CoS value associated with the specified priority.</p>
Step 5	Switch-A /org/qos-policy/egress-policy # set prio sys-class-name	<p>Specifies the system class to be used for the egress policy. The <i>sys-class-name</i> argument can be one of the following class keywords:</p> <ul style="list-style-type: none"> • Fc—Use this priority for QoS policies that control vHBA traffic only. • Platinum—Use this priority for QoS policies that control vNIC traffic only. • Gold—Use this priority for QoS policies that control vNIC traffic only. • Silver—Use this priority for QoS policies that control vNIC traffic only. • Bronze—Use this priority for QoS policies that control vNIC traffic only. • Best Effort—Do not use this priority. It is reserved for the Basic Ethernet traffic lane. If you assign this priority to a QoS policy and configure another system class as CoS 0, Cisco UCS Manager does not default to this system class. It defaults to the priority with CoS 0 for that traffic.
Step 6	Switch-A /org/qos-policy/egress-policy # set rate {line-rate kbps} burst bytes	<p>Specifies the expected average rate of traffic. Traffic that falls under this rate will always conform. The default is line-rate, which equals a value of 10,000,000. The minimum value is 8, and the maximum value is 40,000,000.</p> <p>Rate limiting is supported only on vNICs on the Cisco UCS VIC-1240 Virtual Interface Card and Cisco UCS VIC-1280 Virtual Interface Card. The Cisco UCS M81KR Virtual Interface Card supports rate limiting on both vNICs and vHBAs.</p>
Step 7	Switch-A /org/qos-policy/egress-policy # commit-buffer	Commits the transaction to the system configuration.

Example

The following example creates a QoS policy for vNIC traffic, assigns the platinum system class and sets the rate limit (traffic rate and burst size) for the egress policy, and commits the transaction:

```
Switch-A# scope org /
Switch-A /org # create qos-policy VnicPolicy34
Switch-A /org/qos-policy* # create egress-policy
Switch-A /org/qos-policy/egress-policy* # set prio platinum
Switch-A /org/qos-policy/egress-policy* # set rate 5000000 burst 65000
Switch-A /org/qos-policy/egress-policy* # commit-buffer
Switch-A /org/qos-policy/egress-policy #
```

The following example creates a QoS policy for vHBA traffic, assigns the fc (Fibre Channel) system class and sets the rate limit (traffic rate and burst size) for the egress policy, and commits the transaction:

```
Switch-A# scope org /
Switch-A /org # create qos-policy VhbaPolicy12
Switch-A /org/qos-policy* # create egress-policy
Switch-A /org/qos-policy/egress-policy* # set prio fc
Switch-A /org/qos-policy/egress-policy* # set rate 5000000 burst 65000
Switch-A /org/qos-policy/egress-policy* # commit-buffer
Switch-A /org/qos-policy/egress-policy #
```

What to do next

Include the QoS policy in a vNIC or vHBA template.

Deleting a QoS Policy

If you delete a QoS policy that is in use or you disable a system class that is used in a QoS policy, any vNIC or vHBA that uses that QoS policy is assigned to the Best Effort system class or to the system class with a CoS of 0. In a system that implements multitenancy, Cisco UCS Manager first attempts to find a matching QoS policy in the organization hierarchy.

SUMMARY STEPS

1. UCS-A# **scope org** *org-name*
2. UCS-A /org # **delete qos-policy** *policy-name*
3. UCS-A /org # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope org <i>org-name</i>	Enters organization mode for the specified organization. To enter the root organization mode, type / as the <i>org-name</i> .
Step 2	UCS-A /org # delete qos-policy <i>policy-name</i>	Deletes the specified QoS policy.
Step 3	UCS-A /org # commit-buffer	Commits the transaction to the system configuration.

Example

The following deletes the QoS policy named QosPolicy34 and commits the transaction:

```
UCS-A# scope org /  
UCS-A /org # delete qos-policy QosPolicy34  
UCS-A /org* # commit-buffer  
UCS-A /org #
```

Configuring Flow Control Policies

Flow Control Policy

Flow control policies determine whether the uplink Ethernet ports in a Cisco UCS domain send and receive IEEE 802.3x pause frames when the receive buffer for a port fills. These pause frames request that the transmitting port stop sending data for a few milliseconds until the buffer clears.

For flow control to work between a LAN port and an uplink Ethernet port, you must enable the corresponding receive and send flow control parameters for both ports. For Cisco UCS, the flow control policies configure these parameters.

When you enable the send function, the uplink Ethernet port sends a pause request to the network port if the incoming packet rate becomes too high. The pause remains in effect for a few milliseconds before traffic is reset to normal levels. If you enable the receive function, the uplink Ethernet port honors all pause requests from the network port. All traffic is halted on that uplink port until the network port cancels the pause request.

Because you assign the flow control policy to the port, changes to the policy have an immediate effect on how the port reacts to a pause frame or a full receive buffer.

Configuring a Flow Control Policy

Before you begin

Configure the network port with the corresponding setting for the flow control that you need. For example, if you enable the send setting for flow-control pause frames in the policy, ensure that the receive parameter in the network port is set to on or to desired. If you want the Cisco UCS port to receive flow-control frames, ensure that the send parameter is set to on or to desired on the network port. If you do not want to use flow control, you can set the send and receive parameters on the network port to off.

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **scope flow-control**
3. UCS-A /eth-uplink/flow-control # **create policy** *policy-name*
4. UCS-A /eth-uplink/flow-control/policy # **set prio** *prio-option*
5. UCS-A /eth-uplink/flow-control/policy # **set receive** *receive-option*
6. UCS-A /eth-uplink/flow-control/policy # **set send** *send-option*
7. UCS-A /eth-uplink/flow-control/policy # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # scope flow-control	Enters Ethernet uplink flow control mode.
Step 3	UCS-A /eth-uplink/flow-control # create policy <i>policy-name</i>	Creates the specified flow control policy.
Step 4	UCS-A /eth-uplink/flow-control/policy # set prio <i>prio-option</i>	Specifies one of the following flow control priority options: <ul style="list-style-type: none"> • auto —The Cisco UCS system and the network negotiate whether PPP will be used on this fabric interconnect. • on —PPP is enabled on this fabric interconnect.
Step 5	UCS-A /eth-uplink/flow-control/policy # set receive <i>receive-option</i>	Specifies one of the following flow control receive options: <ul style="list-style-type: none"> • off —Pause requests from the network are ignored and traffic flow continues as normal. • on —Pause requests are honored and all traffic is halted on that uplink port until the network cancels the pause request.
Step 6	UCS-A /eth-uplink/flow-control/policy # set send <i>send-option</i>	Specifies one of the following flow control send options: <ul style="list-style-type: none"> • off —Traffic on the port flows normally regardless of the packet load. • on —The Cisco UCS system sends a pause request to the network if the incoming packet rate becomes too high. The pause remains in effect for a few milliseconds before traffic is reset to normal levels.
Step 7	UCS-A /eth-uplink/flow-control/policy # commit-buffer	Commits the transaction to the system configuration.

Example

The following configures a flow control policy and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope flow-control
UCS-A /eth-uplink/flow-control # create policy FlowControlPolicy23
UCS-A /eth-uplink/flow-control/policy* # set prio auto
UCS-A /eth-uplink/flow-control/policy* # set receive on
UCS-A /eth-uplink/flow-control/policy* # set send on
UCS-A /eth-uplink/flow-control/policy* # commit-buffer
UCS-A /eth-uplink/flow-control/policy #
```

What to do next

Associate the flow control policy with an uplink Ethernet port or port channel.

Deleting a Flow Control Policy

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **scope flow-control**
3. UCS-A /eth-uplink/flow-control # **delete policy** *policy-name*
4. UCS-A /eth-uplink/flow-control # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # scope flow-control	Enters Ethernet uplink flow control mode.
Step 3	UCS-A /eth-uplink/flow-control # delete policy <i>policy-name</i>	Deletes the specified flow control policy.
Step 4	UCS-A /eth-uplink/flow-control # commit-buffer	Commits the transaction to the system configuration.

Example

The following example deletes the flow control policy named FlowControlPolicy23 and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope flow-control
UCS-A /eth-uplink/flow-control # delete policy FlowControlPolicy23
UCS-A /eth-uplink/flow-control* # commit-buffer
UCS-A /eth-uplink/flow-control #
```

Configuring Slow Drain

QoS Slow Drain Device Detection and Mitigation

All data traffic between end devices in the fabric is carried by Fibre Channel services that use link-level, per-hop-based, and buffer-to-buffer flow control. These classes of service do not support end-to-end flow control. When slow devices are attached to the fabric, the end devices do not accept the frames at the configured or negotiated rate. The slow devices lead to an Inter-Switch Link (ISL) credit shortage in the traffic that is destined for these devices, and they congest the links. The credit shortage affects the unrelated flows in the fabric that use the same ISL link even though destination devices do not experience a slow drain.

Similarly, in End-Host Mode, if a server that is directly attached to the Fabric Interconnect receives traffic slowly, it may congest the uplink port shared by other servers. If a slow server is attached to a HIF port on FEX/IOM, it may congest the fabric port and/or uplink port.

Cisco UCS Manager Release 4.0(2) introduces the QoS Slow Drain Detection and Mitigation feature on Cisco UCS 6454 Fabric Interconnects. This feature provides various enhancements that enable you to detect slow drain devices that cause congestion in the network, and also mitigate it. The enhancements are mainly on the edge ports and core ports that connect to the slow drain devices. This is done to minimize the frames stuck condition in the edge and core ports due to slow drain devices that are causing an ISL blockage. To avoid or minimize the stuck condition, you can configure smaller frame timeout for the ports. A smaller frame timeout value helps to alleviate the slow drain condition that affects the fabric by dropping the packets on the edge ports sooner than the time they actually get timed out. This function frees the buffer space in ISL, which can be used by other unrelated flows that do not experience the slow drain condition. Cisco UCS Manager Release 4.1 extends support of this feature to Cisco UCS 64108 Fabric Interconnects.

In this release, slow drain detection and mitigation is supported on the following ports:

- FCoE
- Back-plane

Configuring Slow Drain Detection

SUMMARY STEPS

1. UCS-A# **scope eth-server**
2. UCS-A /eth-server # **scope qos**
3. UCS-A /eth-server/qos # **scope slow-drain**
4. UCS-A /eth-server/qos/slow-drain #**set fcoe-admin-state {disable | enable}**
5. UCS-A /eth-server/qos/slow-drain* # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-server	Enters Ethernet server mode.
Step 2	UCS-A /eth-server # scope qos	Enters Ethernet server QoS mode.
Step 3	UCS-A /eth-server/qos # scope slow-drain	Enters Ethernet server QoS slow drain mode.
Step 4	UCS-A /eth-server/qos/slow-drain # set fcoe-admin-state {disable enable}	Sets the FCoE admin state to one of the following: <ul style="list-style-type: none"> • disable—Slow drain detection is disabled • enable—Slow drain detection is enabled
Step 5	UCS-A /eth-server/qos/slow-drain* # commit-buffer	Commits the transaction to the system configuration.

Example

The following example enables slow drain detection on FCoE ports and commits the transaction:

```

UCS-A# scope eth-server
UCS-A /eth-server # scope qos
UCS-A /eth-server/qos # scope slow-drain
UCS-A /eth-server/qos/slow-drain # set fcoe-admin-state enable
UCS-A /eth-server/qos/slow-drain* # commit-buffer
UCS-A /eth-server/qos/slow-drain #

```

Configuring Slow Drain Timers

While configuring slow drain timeout timers, you can select the timeout value from the list of allowed values. You cannot configure custom timeout values.

SUMMARY STEPS

1. UCS-A# **scope eth-server**
2. UCS-A /eth-server # **scope qos**
3. UCS-A /eth-server/qos # **scope slow-drain**
4. UCS-A /eth-server/qos/slow-drain #**set core-port-timer** {100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900 | 1000}
5. UCS-A /eth-server/qos/slow-drain* #**set edge-port-timer** {100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900 | 1000}
6. UCS-A /eth-server/qos/slow-drain* #**set backplane-port-timer** { 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900 | 1000}
7. UCS-A /eth-server/qos/slow-drain* # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-server	Enters Ethernet server mode.
Step 2	UCS-A /eth-server # scope qos	Enters Ethernet server QoS mode.
Step 3	UCS-A /eth-server/qos # scope slow-drain	Enters Ethernet server QoS slow drain mode.
Step 4	UCS-A /eth-server/qos/slow-drain # set core-port-timer {100 200 300 400 500 600 700 800 900 1000}	Sets the core FCoE port timeout to one of the listed values. The default timeout value is 500 ms.
Step 5	UCS-A /eth-server/qos/slow-drain* # set edge-port-timer {100 200 300 400 500 600 700 800 900 1000}	Sets the edge FCoE port timeout to one of the listed values. The default timeout value is 500 ms.
Step 6	UCS-A /eth-server/qos/slow-drain* # set backplane-port-timer { 200 300 400 500 600 700 800 900 1000}	Sets the backplane port timeout to one of the listed values. The default timeout value is 1000 ms.
Step 7	UCS-A /eth-server/qos/slow-drain* # commit-buffer	Commits the transaction to the system configuration.

Example

The following example configures the slow drain timers and commits the transaction:

```
UCS-A# scope eth-server
UCS-A /eth-server # scope qos
UCS-A /eth-server/qos # scope slow-drain
UCS-A /eth-server/qos/slow-drain # set core-port-timer 500
UCS-A /eth-server/qos/slow-drain* # set edge-port-timer 500
UCS-A /eth-server/qos/slow-drain* # set backplane-port-timer 1000
UCS-A /eth-server/qos/slow-drain* # commit-buffer
UCS-A /eth-server/qos/slow-drain #
```

Displaying Slow Drain Settings

SUMMARY STEPS

1. UCS-A# **scope eth-server**
2. UCS-A /eth-server # **scope qos**
3. UCS-A /eth-server/qos # **show slow-drain**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-server	Enters Ethernet server mode.
Step 2	UCS-A /eth-server # scope qos	Enters Ethernet server QoS mode.
Step 3	UCS-A /eth-server/qos # show slow-drain	Displays QoS slow drain settings.

Example

The following example displays the slow drain settings:

```
UCS-A# scope eth-server
UCS-A /eth-server # scope qos
UCS-A /eth-server/qos # show slow-drain

QoS Slow Drain:
  Admin State for QoS Slow Drain for Physical FCoE Ports: Enabled
  QoS Slow Drain: Timer value for Core Physical FCoE Ports: 100
  QoS Slow Drain: Timer value for Edge Physical FCoE Ports: 100
  QoS Slow Drain: Timer value for Backplane Ports: 1000
UCS-A /eth-server/qos #
```



CHAPTER 9

Port Security

- [Port Security Overview, on page 161](#)
- [Port Security Violations, on page 162](#)
- [Guidelines for Port Security on UCS 6454 Fabric Interconnects, on page 162](#)
- [Configuring Port Security, on page 163](#)

Port Security Overview

The port security feature allows you to restrict input to an interface by limiting and identifying MAC addresses of the workstations that are allowed to access the port. It helps you to control the learning and storing of MAC addresses for each interface. It is used to protect against CAM overflow attacks and rogue equipment, such as hubs and switches, being plugged in. A port security enabled port is called a secure port, and the MAC addresses allowed on that port are called secure MAC addresses. When you assign secure MAC addresses to a secure port, the port does not forward packets with source addresses outside the group of defined addresses. If you limit the number of secure MAC addresses to one and assign a single secure MAC address to a secure port, the workstation attached to that port is assured the full bandwidth of the port.

After you have set the maximum number of secure MAC addresses on a port, you can include secure MAC addresses in an address table in one of these ways:

- Configure all secure MAC addresses by using the `switchport port-security mac-address mac_address` interface configuration command.
- Allow the port to dynamically configure secure MAC addresses with the MAC addresses of connected devices.
- Configure a number of addresses and allow the rest to be dynamically configured.



Note If the port shuts down, all dynamically learned addresses are removed.

- Configure MAC addresses to be sticky. These can be dynamically learned or manually configured, stored in the address table, and added to the running configuration. If these addresses are saved in the configuration file, the interface does not need to dynamically relearn them when the switch restarts. Although sticky secure addresses can be manually configured, it is not recommended.

MAC Learning

After port security is enabled on an interface and a new MAC address is seen on the interface, a security validation is done for the new MAC address. Based on this validation, the MAC address will be added to the address table - either as a normal entry or a drop entry.

Port Security Violations

A port security violation occurs in either of these situations:

- When the maximum number of secure MAC addresses is reached on a secure port and the source MAC address of the ingress traffic is different from any of the identified secure MAC addresses, port security applies the configured violation mode.
- If traffic with a secure MAC address that is configured or learned on one secure port attempts to access another secure port in the same VLAN, port security applies the configured violation mode. This is also known as a MAC move violation.

There are three violation actions for port security. You can configure the port for one of these violation actions:

- **Shutdown**—A port security violation causes the port to shut down immediately.
- **Restrict**—A port security violation restricts data, causes the SecurityViolation counter to increment, and causes an SNMP Trap to be generated. In the Restrict action, learning is disabled on the port after 10 violations. Restrict is the default action for port security violations.
- **Protect**—A port security violation causes data from unknown MAC addresses to be dropped. The SecurityViolation counter is not incremented, and no SNMP Trap is generated.

Guidelines for Port Security on UCS 6454 Fabric Interconnects

The following guidelines apply when you configure port security for UCS 6454 Fabric Interconnect ports:

- Port security can be configured only on NIV ports. It is not supported on BIF ports.
- Only one MAC address per VLAN can be secured for an NIV port.
- For port security violations on virtual interfaces, Restrict is the default violation action.
- MAC learning is disabled on a secure port after 10 violations.
- Secure MAC addresses never age out.
- The maximum number of secure MAC addresses that can be configured are as follows:
 - On a Device—A maximum of 8000 secure MAC addresses in addition to one MAC address per port
 - On an Interface—A maximum of 1000 MAC addresses per interface
 - In a VLAN—Only one secure MAC address per port for a VLAN

Configuring Port Security

To restrict traffic through a port by limiting and identifying MAC addresses of the workstations allowed to access the port, perform this task:

SUMMARY STEPS

1. switch(config)# **interface** *interface_id*
2. switch(config-if)# **switchport mode access**
3. switch(config-if)# [**no**] **switchport port-security**
4. switch(config-if)# **switchport port-security maximum** *value*
5. switch(config-if)# **switchport port-security violation** {**restrict** | **shutdown** | **protect**}
6. switch(config-if)# **switchport port-security mac-address** *mac_address*

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch(config)# interface <i>interface_id</i>	Enters interface configuration mode.
Step 2	switch(config-if)# switchport mode access	Sets the interface mode as access; an interface in the default mode (dynamic desirable) cannot be configured as a secure port.
Step 3	switch(config-if)# [no] switchport port-security	Enables port security on the interface. To return the interface to the default condition as not a secure port, use the no switchport port-security interface configuration command.
Step 4	switch(config-if)# switchport port-security maximum <i>value</i>	Sets the maximum number of secure MAC addresses for the interface. The range is 1 to 1000. To return the interface to the default number of secure MAC addresses, use the no switchport port-security maximum <i>value</i> interface configuration command.
Step 5	switch(config-if)# switchport port-security violation { restrict shutdown protect }	Sets the action to be taken when a security violation is detected. The action can be one of the following: <ul style="list-style-type: none"> • Shutdown—A port security violation causes the port to shut down immediately. • Restrict—A port security violation restricts data, causes the SecurityViolation counter to increment, and causes an SNMP Trap to be generated. In the Restrict action, learning is disabled on the port after 10 violations. Restrict is the default action for port security violations. • Protect—A port security violation causes data from unknown MAC addresses to be dropped. The

	Command or Action	Purpose
		<p>SecurityViolation counter is not incremented, and no SNMP Trap is generated.</p> <p>To return the violation mode to the default condition (restrict), use the no switchport port-security violation {restrict shutdown protect} interface configuration command.</p>
Step 6	switch(config-if)# switchport port-security mac-address <i>mac_address</i>	<p>Enters a secure MAC address for the interface. You can use this command to enter the maximum number of secure MAC addresses. If you configure fewer secure MAC addresses than the maximum, the remaining MAC addresses are dynamically learned.</p> <p>To delete a MAC address from the address table, use the no switchport port-security mac-address <i>mac_address</i> interface configuration command.</p>



CHAPTER 10

Upstream Disjoint Layer-2 Networks

- [Upstream Disjoint Layer-2 Networks, on page 165](#)
- [Guidelines for Configuring Upstream Disjoint L2 Networks, on page 166](#)
- [Upstream Disjoint L2 Networks Pinning Considerations, on page 167](#)
- [Configuring Cisco UCS for Upstream Disjoint L2 Networks, on page 169](#)
- [Assigning Ports and Port Channels to VLANs, on page 170](#)
- [Removing Ports and Port Channels from VLANs, on page 171](#)
- [Viewing Ports and Port Channels Assigned to VLANs, on page 172](#)

Upstream Disjoint Layer-2 Networks

Upstream disjoint layer-2 networks (disjoint L2 networks) are required if you have two or more Ethernet clouds that never connect, but must be accessed by servers or virtual machines located in the same Cisco UCS domain. For example, you could configure disjoint L2 networks if you require one of the following:

- Servers or virtual machines to access a public network and a backup network
- Servers or virtual machines for more than one customer are located in the same Cisco UCS domain, and that need to access the L2 networks for both customers in a multi-tenant system



Note By default, data traffic in Cisco UCS works on a principle of mutual inclusion. All traffic for all VLANs and upstream networks travels along all uplink ports and port channels. If you have upgraded from a release that does not support upstream disjoint layer-2 networks, you must assign the appropriate uplink interfaces to your VLANs, or traffic for those VLANs continues to flow along all uplink ports and port channels.

The configuration for disjoint L2 networks works on a principle of selective exclusion. Traffic for a VLAN that is designated as part of a disjoint network can only travel along an uplink Ethernet port or port channel that is specifically assigned to that VLAN, and is selectively excluded from all other uplink ports and port channels. However, traffic for VLANs that are not specifically assigned to an uplink Ethernet port or port channel can still travel on all uplink ports or port channels, including those that carry traffic for the disjoint L2 networks.

In Cisco UCS, the VLAN represents the upstream disjoint L2 network. When you design your network topology for disjoint L2 networks, you must assign uplink interfaces to VLANs not the reverse.

For information about the maximum number of supported upstream disjoint L2 networks, see the appropriate *Cisco UCS Configuration Limits for Cisco UCS Manager Guide*.

Guidelines for Configuring Upstream Disjoint L2 Networks

When you plan your configuration for upstream disjoint L2 networks, consider the following:

Ethernet Switching Mode Must Be End-Host Mode

Cisco UCS only supports disjoint L2 networks when the Ethernet switching mode of the fabric interconnects is configured for end-host mode. You cannot connect to disjoint L2 networks if the Ethernet switching mode of the fabric interconnects is switch mode.

Symmetrical Configuration Is Recommended for High Availability

If a Cisco UCS domain is configured for high availability with two fabric interconnects, we recommend that both fabric interconnects are configured with the same set of VLANs.

VLAN Validity Criteria Are the Same for Uplink Ethernet Ports and Port Channels

The VLAN used for the disjoint L2 networks must be configured and assigned to an uplink Ethernet port or uplink Ethernet port channel. If the port or port channel does not include the VLAN, Cisco UCS Manager considers the VLAN invalid and does the following:

- Displays a configuration warning in the **Status Details** area for the server.
- Ignores the configuration for the port or port channel and drops all traffic for that VLAN.



Note

The validity criteria are the same for uplink Ethernet ports and uplink Ethernet port channels. Cisco UCS Manager does not differentiate between the two.

Overlapping VLANs Are Not Supported

Cisco UCS does not support overlapping VLANs in disjoint L2 networks. You must ensure that each VLAN only connects to one upstream disjoint L2 domain.

Each vNIC Can Only Communicate with One Disjoint L2 Network

A vNIC can only communicate with one disjoint L2 network. If a server needs to communicate with multiple disjoint L2 networks, you must configure a vNIC for each of those networks.

To communicate with more than two disjoint L2 networks, a server must have a Cisco VIC adapter that supports more than two vNICs.

Appliance Port Must Be Configured with the Same VLAN as Uplink Ethernet Port or Port Channel

For an appliance port to communicate with a disjoint L2 network, you must ensure that at least one uplink Ethernet port or port channel is in the same network and is therefore assigned to the same VLANs that are used by the appliance port. If Cisco UCS Manager cannot identify an uplink Ethernet port or port channel

that includes all VLANs that carry traffic for an appliance port, the appliance port experiences a pinning failure and goes down.

For example, a Cisco UCS domain includes a global VLAN named `vlan500` with an ID of 500. `vlan500` is created as a global VLAN on the uplink Ethernet port. However, Cisco UCS Manager does not propagate this VLAN to appliance ports. To configure an appliance port with `vlan500`, you must create another VLAN named `vlan500` with an ID of 500 for the appliance port. You can create this duplicate VLAN in the **Appliances** node on the **LAN** tab of the Cisco UCS Manager GUI or the **eth-storage** scope in the Cisco UCS Manager CLI. If you are prompted to check for VLAN Overlap, accept the overlap and Cisco UCS Manager creates the duplicate VLAN for the appliance port.

Default VLAN 1 Cannot Be Configured Explicitly on an Uplink Ethernet Port or Port Channel

Cisco UCS Manager implicitly assigns default VLAN 1 to all uplink ports and port channels. Even if you do not configure any other VLANs, Cisco UCS uses default VLAN 1 to handle data traffic for all uplink ports and port channels.



Note

After you configure VLANs in a Cisco UCS domain, default VLAN 1 remains implicitly on all uplink ports and port channels. You cannot explicitly assign default VLAN 1 to an uplink port or port channel, nor can you remove it from an uplink port or port channel.

If you attempt to assign default VLAN 1 to a specific port or port channel, Cisco UCS Manager raises an Update Failed fault.

Therefore, if you configure a Cisco UCS domain for disjoint L2 networks, do not configure any vNICs with default VLAN 1 unless you want all data traffic for that server to be carried on all uplink Ethernet ports and port channels and sent to all upstream networks.

VLANs for Both FIs Must be Concurrently Assigned

When you assign a port to a global VLAN, the VLAN is removed from all of the ports that are not explicitly assigned to the VLAN on both fabric interconnects. The ports on both FIs must be configured at the same time. If the ports are only configured on the first FI, traffic on the second FI will be disrupted.

Upstream Disjoint L2 Networks Pinning Considerations

Communication with an upstream disjoint L2 network requires that you ensure that the pinning is properly configured. Whether you implement soft-pinning or hard-pinning, a VLAN membership mismatch causes traffic for one or more VLANs to be dropped.

Soft-Pinning

Soft-pinning is the default behavior in Cisco UCS. If you plan to implement soft-pinning, you do not need to create LAN pin groups to specify a pin target for a vNIC. Instead, Cisco UCS Manager pins the vNIC to an uplink Ethernet port or port channel according to VLAN membership criteria.

With soft-pinning, Cisco UCS Manager validates data traffic from a vNIC against the VLAN membership of all uplink Ethernet ports and port channels. If you have configured disjoint L2 networks, Cisco UCS Manager must be able to find an uplink Ethernet port or port channel that is assigned to all VLANs on the vNIC. If no

uplink Ethernet port or port channel is configured with all VLANs on the vNIC, Cisco UCS Manager does the following:

- Brings the link down.
- Drops the traffic for all of the VLANs on the vNIC.
- Raises the following faults:
 - Link Down
 - VIF Down

Cisco UCS Manager does not raise a fault or warning about the VLAN configuration.

For example, a vNIC on a server is configured with VLANs 101, 102, and 103. Interface 1/3 is assigned only to VLAN 102. Interfaces 1/1 and 1/2 are not explicitly assigned to a VLAN, which makes them available for traffic on VLANs 101 and 103. As a result of this configuration, the Cisco UCS domain does not include a border port interface that can carry traffic for all three VLANs for which the vNIC is configured. As a result, Cisco UCS Manager brings down the vNIC, drops traffic for all three VLANs on the vNIC, and raises the Link Down and VIF Down faults.

hard-pinning

hard-pinning occurs when you use LAN pin groups to specify the pinning target for the traffic intended for the disjoint L2 networks. In turn, the uplink Ethernet port or port channel that is the pinning target must be configured to communicate with the appropriate disjoint L2 network.

With hard-pinning, Cisco UCS Manager validates data traffic from a vNIC against the VLAN membership of all uplink Ethernet ports and port channels, and validates the LAN pin group configuration to ensure it includes the VLAN and the uplink Ethernet port or port channel. If the validation fails at any point, Cisco UCS Manager does the following:

- Raises a Pinning VLAN Mismatch fault with a severity of Warning.
- Drops traffic for the VLAN.
- Does not bring the link down, so that traffic for other VLANs can continue to flow along it.

For example, if you want to configure hard-pinning for an upstream disjoint L2 network that uses VLAN 177, do the following:

- Create a LAN pin group with the uplink Ethernet port or port channel that carries the traffic for the disjoint L2 network.
- Configure at least one vNIC in the service profile with VLAN 177 and the LAN pin group.
- Assign VLAN 177 to an uplink Ethernet port or port channel included in the LAN pin group

If the configuration fails at any of these three points, then Cisco UCS Manager warns of a VLAN mismatch for VLAN 177 and drops the traffic for that VLAN only.



Note If changes are made to soft-pinning configurations resulting in vNIC VLANs not resolving with disjoint L2 uplink, a warning dialog box is displayed. The warning dialog box allows you to proceed with your configuration or cancel it. If you decide to proceed with the mis-configuration, you will experience a reduction in server traffic performance.

Configuring Cisco UCS for Upstream Disjoint L2 Networks

When you configure a Cisco UCS domain to connect with upstream disjoint L2 networks, you need to ensure that you complete all of the following steps.

Before you begin

Before you begin this configuration, ensure that the ports on the fabric interconnects are properly cabled to support your disjoint L2 networks configuration.

SUMMARY STEPS

1. Configure Ethernet switching mode for both fabric interconnects in Ethernet End-Host Mode.
2. Configure the ports and port channels that you require to carry traffic for the disjoint L2 networks.
3. (Optional) Configure the LAN pin groups required to pin the traffic for the appropriate uplink Ethernet ports or port channels.
4. Create one or more VLANs.
5. Assign the desired ports or port channels to the VLANs for the disjoint L2 networks.
6. Ensure that the service profiles for all servers that need to communicate with the disjoint L2 networks include the correct LAN connectivity configuration. This configuration ensures that the vNICs direct the traffic to the appropriate VLAN.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Configure Ethernet switching mode for both fabric interconnects in Ethernet End-Host Mode.	The Ethernet switching mode must be in End-Host Mode for Cisco UCS to be able to communicate with upstream disjoint L2 networks. See LAN Ports and Port Channels , on page 21.
Step 2	Configure the ports and port channels that you require to carry traffic for the disjoint L2 networks.	
Step 3	(Optional) Configure the LAN pin groups required to pin the traffic for the appropriate uplink Ethernet ports or port channels.	See Configuring a LAN Pin Group , on page 141.
Step 4	Create one or more VLANs.	These can be named VLANs or private VLANs. For a cluster configuration, we recommend that you create the VLANs accessible to both fabric interconnects. See VLANs , on page 107 and Upstream Disjointed Layer-2 Networks , on page 165

	Command or Action	Purpose
Step 5	Assign the desired ports or port channels to the VLANs for the disjoint L2 networks.	When this step is complete, traffic for these VLANs is sent through the trunks for the assigned ports and/or port channels.
Step 6	Ensure that the service profiles for all servers that need to communicate with the disjoint L2 networks include the correct LAN connectivity configuration. This configuration ensures that the vNICs direct the traffic to the appropriate VLAN.	You can complete this configuration through one or more vNIC templates, or when you configure the networking options for the service profile. For more information about vNIC templates and service profiles, see the <i>Cisco UCS Manager Storage Management Guide</i> .

Assigning Ports and Port Channels to VLANs

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **scope vlan** *vlan-name*
3. UCS-A /eth-uplink/vlan # **create member-port** *fabric-interconnect slot-id port-id*
4. UCS-A /eth-uplink/vlan # **create member-port-channel** *fabric-interconnect member-port-chan-id*
5. UCS-A /eth-uplink/vlan # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # scope vlan <i>vlan-name</i>	Enters Ethernet uplink VLAN mode for the specified VLAN.
Step 3	UCS-A /eth-uplink/vlan # create member-port <i>fabric-interconnect slot-id port-id</i>	Assigns the specified VLAN to the specified uplink Ethernet port.
Step 4	UCS-A /eth-uplink/vlan # create member-port-channel <i>fabric-interconnect member-port-chan-id</i>	Assigns the specified VLAN to the specified uplink Ethernet port channel.
Step 5	UCS-A /eth-uplink/vlan # commit-buffer	Commits the transaction to the system configuration. After a port or port channel is assigned to one or more VLANs, it is removed from all other VLANs.

Example

The following example assigns uplink Ethernet ports to a named VLAN called VLAN100 on fabric interconnect A and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope vlan VLAN100
UCS-A /eth-uplink/vlan # create member-port a 2
```

```
UCS-A /eth-uplink/vlan # create member-port a 4
UCS-A /eth-uplink/vlan* # commit-buffer
UCS-A /eth-uplink/vlan #
```

Removing Ports and Port Channels from VLANs

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **scope vlan** *vlan-name*
3. UCS-A /eth-uplink/vlan # **delete member-port** *fabric-interconnect slot-id port-id*
4. UCS-A /eth-uplink/vlan # **delete member-port-channel** *fabric-interconnect member-port-chan-id*
5. UCS-A /eth-uplink/vlan # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # scope vlan <i>vlan-name</i>	Enters Ethernet uplink VLAN mode for the specified VLAN.
Step 3	UCS-A /eth-uplink/vlan # delete member-port <i>fabric-interconnect slot-id port-id</i>	Deletes the specified Uplink Ethernet member port assignment from the VLAN.
Step 4	UCS-A /eth-uplink/vlan # delete member-port-channel <i>fabric-interconnect member-port-chan-id</i>	Deletes the specified Uplink Ethernet port channel assignment from the VLAN.
Step 5	UCS-A /eth-uplink/vlan # commit-buffer	Commits the transaction to the system configuration. Important If you remove all port or port channel interfaces from a VLAN, the VLAN returns to the default behavior and data traffic on that VLAN flows on all uplink ports and port channels. Based on the configuration in the Cisco UCS domain, this default behavior can cause Cisco UCS Manager to drop traffic for that VLAN. To avoid this occurrence, Cisco recommends that you assign at least one interface to the VLAN or delete the VLAN.

Example

The following example deletes the association between uplink Ethernet port 2 on fabric interconnect A and the named VLAN called MyVLAN and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope vlan MyVLAN
UCS-A /eth-uplink/vlan # delete member-port a 2
```

```
UCS-A /eth-uplink/vlan* # commit-buffer
UCS-A /eth-uplink/vlan #
```

Viewing Ports and Port Channels Assigned to VLANs

SUMMARY STEPS

1. UCS-A# **scope eth-uplink**
2. UCS-A /eth-uplink # **scope vlan** *vlan-name*
3. UCS-A /eth-uplink/vlan # **show member-port** [detail | expand]
4. UCS-A /eth-uplink/vlan # **show member-port-channel** [detail | expand]
5. UCS-A /eth-uplink/vlan # **commit-buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	UCS-A# scope eth-uplink	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # scope vlan <i>vlan-name</i>	Enters Ethernet uplink VLAN mode for the specified VLAN.
Step 3	UCS-A /eth-uplink/vlan # show member-port [detail expand]	Shows member ports assigned to the specified VLAN.
Step 4	UCS-A /eth-uplink/vlan # show member-port-channel [detail expand]	Shows member port channels assigned to the specified VLAN.
Step 5	UCS-A /eth-uplink/vlan # commit-buffer	Commits the transaction to the system configuration.

Example

The following example displays the full details for uplink Ethernet ports assigned to a named VLAN called MyVLAN:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope vlan MyVLAN
UCS-A /eth-uplink/vlan # show member-port detail
Member Port:
  Fabric ID: A
  Slot ID: 1
  Port ID: 2
  Mark Native Vlan: No
UCS-A /eth-uplink/vlan #
```



CHAPTER 11

Network-Related Policies

- [Configuring vNIC Templates, on page 173](#)
- [Configuring Adapter Policies, on page 181](#)
- [Configuring the Default vNIC Behavior Policy, on page 198](#)
- [Configuring LAN Connectivity Policies, on page 199](#)
- [Configuring Network Control Policies, on page 207](#)
- [Configuring Multicast Policies, on page 210](#)
- [Configuring LACP Policies, on page 213](#)
- [Configuring UDLD Link Policies, on page 214](#)
- [Configuring VMQ and VMMQ Connection Policies, on page 218](#)
- [NetQueue, on page 225](#)

Configuring vNIC Templates

vNIC Template

The vNIC LAN connectivity policy defines how a vNIC on a server connects to the LAN.

Cisco UCS Manager does not automatically create a VM-FEX port profile with the correct settings when you create a vNIC template. If you want to create a VM-FEX port profile, you must configure the target of the vNIC template as a VM. You must include this policy in a service profile for it to take effect.

You can select VLAN groups in addition to any individual VLAN while creating a vNIC template.



Note

If your server has two Emulex or QLogic NICs (Cisco UCS CNA M71KR-E or Cisco UCS CNA M71KR-Q), you must configure vNIC policies for both adapters in your service profile to get a user-defined MAC address for both NICs. If you do not configure policies for both NICs, Windows still detects both of them in the PCI bus. Then because the second eth is not part of your service profile, Windows assigns it a hardware MAC address. If you then move the service profile to a different server, Windows sees additional NICs because one NIC did not have a user-defined MAC address.

Creating a vNIC Template

Before you begin

This policy requires that one or more of the following resources already exist in the system:

- Named VLAN
- MAC pool
- QoS policy
- LAN pin group
- Statistics threshold policy

SUMMARY STEPS

1. In the **Navigation** pane, click **LAN**.
2. Expand **LAN > Policies**.
3. Expand the node for the organization where you want to create the policy.
4. Right-click the **vNIC Templates** node and choose **Create vNIC Template**.
5. In the **Create vNIC Template** dialog box:
6. Click **OK**.

DETAILED STEPS

- Step 1** In the **Navigation** pane, click **LAN**.
- Step 2** Expand **LAN > Policies**.
- Step 3** Expand the node for the organization where you want to create the policy.
If the system does not include multitenancy, expand the **root** node.
- Step 4** Right-click the **vNIC Templates** node and choose **Create vNIC Template**.
- Step 5** In the **Create vNIC Template** dialog box:
- a) In the **General** area, complete the following fields:

Name	Description
Name field	The name of the vNIC template. This name can be between 1 and 16 alphanumeric characters. You cannot use spaces or any special characters other than - (hyphen), _ (underscore), : (colon), and . (period), and you cannot change this name after the object is saved.
Description field	A user-defined description of the template. Enter up to 256 characters. You can use any characters or spaces except ` (accent mark), \ (backslash), ^ (carat), " (double quote), = (equal sign), > (greater than), < (less than), or ' (single quote).

Name	Description
Fabric ID field	<p>The fabric interconnect associated with the component.</p> <p>If you want vNICs created from this template to be able to access the second fabric interconnect if the default one is unavailable, check the Enable Failover check box.</p> <p>Note Do not enable vNIC fabric failover under the following circumstances:</p> <ul style="list-style-type: none"> • If the Cisco UCS domain is running in Ethernet switch mode. vNIC fabric failover is not supported in that mode. If all Ethernet uplinks on one fabric interconnect fail, the vNICs do not fail over to the other. • If you plan to associate one or more vNICs created from this template to a server with an adapter that does not support fabric failover, such as the Cisco UCS 82598KR-CI 10-Gigabit Ethernet Adapter. If so, Cisco UCS Manager generates a configuration fault when you associate the service profile with the server.
Redundancy Type	<p>The Redundancy type that you choose initiates a fabric failover using vNIC/HBA redundancy pairs.</p> <ul style="list-style-type: none"> • Primary Template— Creates configurations that can be shared with the Secondary template. Any other shared changes on the Primary template are automatically synchronized to the Secondary template. • Secondary Template— All shared configurations are inherited from the Primary template. • No Redundancy— Legacy vNIC/vHBA template behavior. Select this option if you do not want to use redundancy.
Target list box	<p>A list of the possible targets for vNICs created from this template. The target you choose determines whether or not Cisco UCS Manager automatically creates a VM-FEX port profile with the appropriate settings for the vNIC template. This can be one of the following:</p> <ul style="list-style-type: none"> • Adapter—The vNICs apply to all adapters. No VM-FEX port profile is created if you choose this option. • VM—The vNICs apply to all virtual machines. A VM-FEX port profile is created if you choose this option.
Template Type field	<ul style="list-style-type: none"> • Initial Template: vNICs created from this template are not updated if the template changes. • Updating Template: vNICs created from this template are updated if the template changes.

- b) In the **VLANs** area, use the table to select the VLAN to assign to vNICs created from this template. The table contains the following columns:

Name	Description
Select column	Check the check box in this column for each VLAN that you want to use. Note VLANs and PVLANS can not be assigned to the same vNIC.
Name column	The name of the VLAN.
Native VLAN column	To designate one of the VLANs as the native VLAN, click the radio button in this column.

- c) In the **VLAN Groups** area, use the table to select the VLAN group to assign to vNICs created from this template. The table contains the following columns:

Name	Description
Select column	Check the check box in this column for each VLAN Group that you want to use.
Name column	The name of the VLAN Group.

- d) In the **Policies** area, complete the following fields:

Name	Description
CDN Source field	<p>This can be one of the following options:</p> <ul style="list-style-type: none"> • vNIC Name <ul style="list-style-type: none"> — Uses the vNIC template name of the vNIC instance as the CDN name. This is the default option. • User Defined <ul style="list-style-type: none"> — Displays the CDN Name field for you to enter a user-defined CDN name for the vNIC template. <p>Refer to the <i>Cisco UCS Manager Server Management Guide</i> for more information on Consistent Device Naming.</p>

Name	Description
MTU field	<p>The maximum transmission unit, or packet size, that vNICs created from this vNIC template should use.</p> <p>Enter an integer between 1500 and 9000.</p> <p>Note If the vNIC template has an associated QoS policy, the MTU specified here must be equal to or less than the MTU specified in the associated QoS system class. If this MTU value exceeds the MTU value in the QoS system class, packets may be dropped during data transmission.</p> <p>For VIC 14xx adapters, you can change the MTU size of the vNIC from the host interface settings. When the Overlay network is configured, make sure that the new value is equal to or less than the MTU specified in the associated QoS system class or packets could be dropped during data transmission.</p>
MAC Pool drop-down list	The MAC address pool that vNICs created from this vNIC template should use.
QoS Policy drop-down list	The quality of service policy that vNICs created from this vNIC template should use.
Network Control Policy drop-down list	The network control policy that vNICs created from this vNIC template should use.
Pin Group drop-down list	The LAN pin group that vNICs created from this vNIC template should use.
Stats Threshold Policy drop-down list	The statistics collection policy that vNICs created from this vNIC template should use.

Step 6 Click **OK**.

What to do next

Include the vNIC template in a service profile.

Creating vNIC Template Pairs

- Step 1** In the Navigation pane, click the **LAN** tab. On the **LAN** tab, expand **LAN > Policies**.
- Step 2** Expand the node for the organization where you want to create the policy. If the system does not include multi-tenancy, expand the root node.
- Step 3** Right-click the **vNIC Templates** node and choose **Create vNIC Template**. In the **Create vNIC Template** dialog box, assign a **Name**, **Description**, and select the **Fabric ID** for the template.
- Step 4** Select the **Redundancy Type** as **Primary** or **Secondary** or **No Redundancy**. See the redundancy type descriptions below.

Step 5 Select the **Peer Redundancy Template**—to choose the name of the corresponding **Primary** or **Secondary** redundancy template to perform the template pairing from the **Primary** or **Secondary** redundancy template.

- **Primary**—Creates configurations that can be shared with the Secondary template. Any other shared changes on the Primary template are automatically synchronized to the Secondary template.

- **VLANS**
- **Template Type**
- **MTU**
- **Network Control Policies**
- **Connection Policies**
- **QoS Policy**
- **Stats Threshold Policy**

Following is a list of non-shared configurations:

- **Fabric ID**

Note The Fabric ID must be mutually exclusive. If you assign the Primary template to Fabric A, then Fabric B is automatically assigned to the Secondary template as part of the synchronization from the Primary template.

- **CDN Source**
- **MAC Pool**
- **Description**
- **Pin Group Policy**

- **Secondary**—

All shared configurations are inherited from the Primary template.

- **No Redundancy**—

Legacy vNIC template behavior.

Step 6 Click **OK**.

What to do next

After you create the vNIC redundancy template pair, you can use the redundancy template pair to create redundancy vNIC pairs for any service profile in the same organization or sub-organization.

Undo vNIC Template Pairs

You can undo the vNIC template pair by changing the Peer Redundancy Template so that there is no peer template for the Primary or the Secondary template. When you undo a vNIC template pair, the corresponding vNIC pairs also becomes undone.

SUMMARY STEPS

1. Select **not set** from the **Peer Redundancy Template** drop-down list to undo the pairing between the peer Primary or Secondary redundancy template used to perform the template pairing. You can also select **None** as the **Redundancy Type** to undo the pairing.

DETAILED STEPS

Select **not set** from the **Peer Redundancy Template** drop-down list to undo the pairing between the peer Primary or Secondary redundancy template used to perform the template pairing. You can also select **None** as the **Redundancy Type** to undo the pairing.

Note If you delete one template in a pair, you are prompt to delete the other template in the pair. If you do not delete the other template in the pair, that template resets its peer reference and retains its redundancy type.

Binding a vNIC to a vNIC Template

You can bind a vNIC associated with a service profile to a vNIC template. When you bind the vNIC to a vNIC template, Cisco UCS Manager configures the vNIC with the values defined in the vNIC template. If the existing vNIC configuration does not match the vNIC template, Cisco UCS Manager reconfigures the vNIC. You can only change the configuration of a bound vNIC through the associated vNIC template. You cannot bind a vNIC to a vNIC template if the service profile that includes the vNIC is already bound to a service profile template.



Important

If the vNIC is reconfigured when you bind it to a template, Cisco UCS Manager reboots the server associated with the service profile.

SUMMARY STEPS

1. In the **Navigation** pane, click **Servers**.
2. Expand **Servers > Service Profiles**.
3. Expand the node for the organization that includes the service profile with the vNIC you want to bind.
4. Expand **Service_Profile_Name > vNICs**.
5. Click the vNIC you want to bind to a template.
6. In the **Work** pane, click the **General** tab.
7. In the **Actions** area, click **Bind to a Template**.
8. In the **Bind to a vNIC Template** dialog box, do the following:
9. In the warning dialog box, click **Yes** to acknowledge that Cisco UCS Manager may need to reboot the server if the binding causes the vNIC to be reconfigured.

DETAILED STEPS

Step 1 In the **Navigation** pane, click **Servers**.

- Step 2** Expand **Servers > Service Profiles**.
- Step 3** Expand the node for the organization that includes the service profile with the vNIC you want to bind.
If the system does not include multi-tenancy, expand the **root** node.
- Step 4** Expand *Service_Profile_Name* > vNICs.
- Step 5** Click the vNIC you want to bind to a template.
- Step 6** In the **Work** pane, click the **General** tab.
- Step 7** In the **Actions** area, click **Bind to a Template**.
- Step 8** In the **Bind to a vNIC Template** dialog box, do the following:
- From the **vNIC Template** drop-down list, choose the template to which you want to bind the vNIC.
 - Click **OK**.
- Step 9** In the warning dialog box, click **Yes** to acknowledge that Cisco UCS Manager may need to reboot the server if the binding causes the vNIC to be reconfigured.
-

Unbinding a vNIC from a vNIC Template

SUMMARY STEPS

1. In the **Navigation** pane, click **Servers**.
2. Expand **Servers > Service Profiles**.
3. Expand the node for the organization that includes the service profile with the vNIC you want to unbind.
4. Expand *Service_Profile_Name* > vNICs.
5. Click the vNIC you want to unbind from a template.
6. In the **Work** pane, click the **General** tab.
7. In the **Actions** area, click **Unbind from a Template**.
8. If a confirmation dialog box displays, click **Yes**.

DETAILED STEPS

-
- Step 1** In the **Navigation** pane, click **Servers**.
- Step 2** Expand **Servers > Service Profiles**.
- Step 3** Expand the node for the organization that includes the service profile with the vNIC you want to unbind.
If the system does not include multi-tenancy, expand the **root** node.
- Step 4** Expand *Service_Profile_Name* > vNICs.
- Step 5** Click the vNIC you want to unbind from a template.
- Step 6** In the **Work** pane, click the **General** tab.
- Step 7** In the **Actions** area, click **Unbind from a Template**.
- Step 8** If a confirmation dialog box displays, click **Yes**.
-

Deleting a vNIC Template

SUMMARY STEPS

1. In the **Navigation** pane, click **LAN**.
2. Expand **LAN > Policies > *Organization_Name***.
3. Expand the **vNIC Templates** node.
4. Right-click the policy you want to delete and choose **Delete**.
5. If a confirmation dialog box displays, click **Yes**.

DETAILED STEPS

-
- | | |
|---------------|--|
| Step 1 | In the Navigation pane, click LAN . |
| Step 2 | Expand LAN > Policies > <i>Organization_Name</i> . |
| Step 3 | Expand the vNIC Templates node. |
| Step 4 | Right-click the policy you want to delete and choose Delete . |
| Step 5 | If a confirmation dialog box displays, click Yes . |
-

Configuring Adapter Policies

Ethernet and Fibre Channel Adapter Policies

These policies govern the host-side behavior of the adapter, including how the adapter handles traffic. For example, you can use these policies to change default settings for the following:

- Queues
- Interrupt handling
- Performance enhancement
- RSS hash
- Failover in a cluster configuration with two fabric interconnects

**Note**

For Fibre Channel adapter policies, the values displayed by Cisco UCS Manager may not match those displayed by applications such as QLogic SANsurfer. For example, the following values may result in an apparent mismatch between SANsurfer and Cisco UCS Manager:

- **Max LUNs Per Target**—SANsurfer has a maximum of 256 LUNs and does not display more than that number. Cisco UCS Manager supports a higher maximum number of LUNs. This parameter is applicable only for FC-Initiator.
- **Link Down Timeout**—In SANsurfer, you configure the timeout threshold for link down in seconds. In Cisco UCS Manager, you configure this value in milliseconds. Therefore, a value of 5500 ms in Cisco UCS Manager displays as 5s in SANsurfer.
- **Max Data Field Size**—SANsurfer has allowed values of 512, 1024, and 2048. Cisco UCS Manager allows you to set values of any size. Therefore, a value of 900 in Cisco UCS Manager displays as 512 in SANsurfer.
- **LUN Queue Depth**—The LUN queue depth setting is available for Windows system FC adapter policies. Queue depth is the number of commands that the HBA can send and receive in a single transmission per LUN. Windows Storport driver sets this to a default value of 20 for physical miniports and to 250 for virtual miniports. This setting adjusts the initial queue depth for all LUNs on the adapter. Valid range for this value is 1 - 254. The default LUN queue depth is 20. This feature only works with Cisco UCS Manager version 3.1(2) and higher. This parameter is applicable only for FC-Initiator.
- **IO TimeOut Retry**—When the target device does not respond to an IO request within the specified timeout, the FC adapter cancels the pending command then resends the same IO after the timer expires. The FC adapter valid range for this value is 1 - 59 seconds. The default IO retry timeout is 5 seconds. This feature only works with Cisco UCS Manager version 3.1(2) and higher.

Operating System Specific Adapter Policies

By default, Cisco UCS provides a set of Ethernet adapter policies and Fibre Channel adapter policies. These policies include the recommended settings for each supported server operating system. Operating systems are sensitive to the settings in these policies. Storage vendors typically require non-default adapter settings. You can find the details of these required settings on the support list provided by those vendors.

**Important**

We recommend that you use the values in these policies for the applicable operating system. Do not modify any of the values in the default policies unless directed to do so by Cisco Technical Support.

However, if you are creating an Ethernet adapter policy for an OS (instead of using the default adapter policy), you must use the following formulas to calculate values that work for that OS.

Depending on the UCS firmware, your driver interrupt calculations may be different. Newer UCS firmware uses a calculation that differs from previous versions. Later driver release versions on Linux operating systems now use a different formula to calculate the Interrupt Count. In this formula, the Interrupt Count is the maximum of either the Transmit Queue or the Receive Queue plus 2.

Interrupt Count in Linux Adapter Policies

Drivers on Linux operating systems use differing formulas to calculate the Interrupt Count, depending on the eNIC driver version. The UCS 3.2 release increased the number of Tx and Rx queues for the eNIC driver from 8 to 256 each.

Use one of the following strategies, according to your driver version.

For Linux drivers before the UCS 3.2 firmware release, use the following formula to calculate the Interrupt Count.

Completion Queues = Transmit Queues + Receive Queues

Interrupt Count = (Completion Queues + 2) rounded up to nearest power of 2

For example, if Transmit Queues = 1 and Receive Queues = 8 then:

Completion Queues = 1 + 8 = 9

Interrupt Count = (9 + 2) rounded up to the nearest power of 2 = 16

On drivers for UCS firmware release 3.2 and higher, the Linux eNIC drivers use the following formula to calculate the Interrupt Count.

Interrupt Count = (#Tx or Rx Queues) + 2

For example:

Interrupt Count wq = 32, rq = 32, cq = 64 - then Interrupt Count = Max(32, 32) + 2 = 34

Interrupt Count wq = 64, rq = 8, cq = 72 - then Interrupt Count = Max(64, 8) + 2 = 66

Interrupt Count wq = 1, rq = 16, cq = 17 - then Interrupt count = Max(1, 16) + 2 = 18

NVMe over Fabrics using Fibre Channel

The NVMe Express (NVMe) interface allows host software to communicate with a non-volatile memory subsystem. This interface is optimized for Enterprise non-volatile storage, which is typically attached as a register level interface to the PCI Express (PCIe) interface.

NVMe over Fabrics using Fibre Channel (FC-NVMe) defines a mapping protocol for applying the NVMe interface to Fibre Channel. This protocol defines how Fibre Channel services and specified Information Units (IUs) are used to perform the services defined by NVMe over a Fibre Channel fabric. NVMe initiators can access and transfer information to NVMe targets over Fibre Channel.

FC-NVMe combines the advantages of Fibre Channel and NVMe. You get the improved performance of NVMe along with the flexibility and the scalability of the shared storage architecture. Cisco UCS Manager Release 4.0(2) supports NVMe over Fabrics using Fibre Channel on UCS VIC 14xx adapters.

Cisco UCS Manager provides the recommended FcNVMe adapter policies in the list of pre-configured adapter policies. To create a new FcNVMe adapter policy, follow the steps in the *Creating a Fibre Channel Adapter Policy* section.

NVMe over Fabrics Using RDMA

NVMe over Fabrics (NVMeoF) is a communication protocol that allows one computer to access NVMe namespaces available on another computer. NVMeoF is similar to NVMe, but differs in the network-related steps involved in using the NVMeoF storage devices. The commands for discovering, connecting, and disconnecting a NVMeoF storage device are integrated into the **nvme** utility provided in Linux..

The NVMeoF fabric that Cisco supports is RDMA over Converged Ethernet version 2 (RoCE v2). RoCE v2 is a fabric protocol that runs over UDP. It requires a no-drop policy.

The eNIC RDMA driver works in conjunction with the eNIC driver, which must be loaded first when configuring NVMeoF.

Cisco UCS Manager provides the default Linux-NVMe-RoCE adapter policy for creating NVMe RoCE v2 interfaces. Do not use the default Linux adapter policy. For complete information on configuring RoCE v2 over NVMeoF, refer to the *Cisco UCS Manager Configuration Guide for RDMA over Converged Ethernet (RoCE) v2*.

NVMeoF using RDMA is supported on M5 B-Series or C-Series Servers with Cisco UCS VIC 1400 Series adapters.

Accelerated Receive Flow Steering

Accelerated Receive Flow Steering (ARFS) is hardware-assisted receive flow steering that can increase CPU data cache hit rate by steering kernel level processing of packets to the CPU where the application thread consuming the packet is running.

Using ARFS can improve CPU efficiency and reduce traffic latency. Each receive queue of a CPU has an interrupt associated with it. You can configure the Interrupt Service Routine (ISR) to run on a CPU. The ISR moves the packet from the receive queue to the backlog of one of the current CPUs, which processes the packet later. If the application is not running on this CPU, the CPU must copy the packet to non-local memory, which adds to latency. ARFS can reduce this latency by moving that particular stream to the receive queue of the CPU on which the application is running.

ARFS is disabled by default and can be enabled through Cisco UCS Manager. To configure ARFS, do the following:

1. Create an adapter policy with ARFS enabled.
2. Associate the adapter policy with a service profile.
3. Enable ARFS on a host.
 - a. Turn off Interrupt Request Queue (IRQ) balance.
 - b. Associate IRQ with different CPUs.
 - c. Enable ntuple by using ethtool.

Guidelines and Limitations for Accelerated Receive Flow Steering

- ARFS supports 64 filters per vNIC
- ARFS is supported on the following adapters:
 - Cisco UCS VIC 12XX
 - Cisco UCS VIC 13XX
 - Cisco UCS VIC 14XX
- ARFS is supported on the following Operating Systems:
 - Red Hat Enterprise Linux 6.5 and higher versions
 - Red Hat Enterprise Linux 7.0 and higher versions
 - Red Hat Enterprise Linux 8.0 and higher versions

- SUSE Linux Enterprise Server 11 SP2 and higher versions
- SUSE Linux Enterprise Server 12 SP1 - SP3
- SUSE Linux Enterprise Server 15 and higher versions
- Ubuntu 14.04.2 and higher versions

Interrupt Coalescing

Adapters typically generate a large number of interrupts that a host CPU must service. Interrupt coalescing reduces the number of interrupts serviced by the host CPU. This is done by interrupting the host only once for multiple occurrences of the same event over a configurable coalescing interval.

When interrupt coalescing is enabled for receive operations, the adapter continues to receive packets, but the host CPU does not immediately receive an interrupt for each packet. A coalescing timer starts when the first packet is received by the adapter. When the configured coalescing interval times out, the adapter generates one interrupt with the packets received during that interval. The NIC driver on the host then services the multiple packets that are received. Reduction in the number of interrupts generated reduces the time spent by the host CPU on context switches. This means that the CPU has more time to process packets, which results in better throughput and latency.

Adaptive Interrupt Coalescing

Due to the coalescing interval, the handling of received packets adds to latency. For small packets with a low packet rate, this latency increases. To avoid this increase in latency, the driver can adapt to the pattern of traffic flowing through it and adjust the interrupt coalescing interval for a better response from the server.

Adaptive interrupt coalescing (AIC) is most effective in connection-oriented low link utilization scenarios including email server, databases server, and LDAP server. It is not suited for line-rate traffic.

Guidelines and Limitations for Adaptive Interrupt Coalescing

- Adaptive Interrupt Coalescing (AIC) does not provide any reduction in latency when the link utilization is more than 80 percent.
- Enabling AIC disables static coalescing.
- AIC is supported on the following Operating Systems:
 - Red Hat Enterprise Linux 6.4 and higher versions
 - SUSE Linux Enterprise Server 11 SP2 and higher versions
 - XenServer 6.5 and higher versions
 - Ubuntu 14.04.2 and higher versions

RDMA Over Converged Ethernet Overview

Remote Direct Memory Access (RDMA) improves performance by enabling direct data exchange in and out of a server. NVMe over Ethernet (NVMeoF) support for RDMA provides faster access to NVMe namespaces on another computer. RDMA over Converged Ethernet (RoCE) allows direct memory access over an Ethernet network. RoCE is a link layer protocol, and hence, it allows communication between any two hosts in the same Ethernet broadcast domain. RoCE delivers superior performance compared to traditional network socket

implementations because of lower latency, lower CPU utilization and higher utilization of network bandwidth. Windows 2012 R2 and later versions use RDMA for accelerating and improving the performance of SMB file sharing and Live Migration.

Cisco UCS Manager supports RoCE for Microsoft SMB Direct. It sends additional configuration information to the adapter while creating or modifying an Ethernet adapter policy. Basic RoCE is also referred to as RoCE v1, and is supported on UCS Manager releases from UCS Manager 2.2(4b) to 4.1(1a).

With Cisco UCS Manager 4.1(1a) and later releases, the RoCE v2 protocol is used.

RDMA Over Converged Ethernet (RoCE) v2

RDMA over Converged Ethernet version 2 (RoCE v2) is an *internet layer* protocol, which means that RoCE v2 packets can be routed. RoCE v2 allows direct memory access over the network by encapsulating an Infiniband (IB) transport packet over Ethernet.

The RoCE v2 protocol exists on top of either the UDP/IPv4 or the UDP/IPv6 protocol. The UDP destination port number 4791 has been reserved for RoCE v2. Since RoCE v2 packets are routable, the RoCE v2 protocol is sometimes called Routable RoCE.

RoCE v2 is supported on the Windows and Linux platforms.

Guidelines and Limitations for SMB Direct with RoCE v1

RoCE v1 for SMB Direct is supported on UCS Manager releases 2.2(4b) up to UCS Manager 4.1(1a). The RoCE v2 protocol is used with UCS Manager 4.1(1x) and later releases.



Note

RoCE v1 is not supported with any fourth generation Cisco UCS VIC 1400 Series adapters.

- Microsoft SMB Direct with RoCE v1 is supported on Microsoft Windows, Release 2012 R2 for Cisco UCS Manager release 2.2(4) and later releases.
- For Microsoft SMB Direct with RoCE support on Microsoft Windows 2016 for Cisco UCS Manager release, check [UCS Hardware and Software Compatibility](#).
- Microsoft SMB Direct with RoCE v1 is supported only with third generation Cisco UCS VIC 1340, 1380, 1385, 1387 adapters. Second generation UCS VIC 12XX adapters are not supported.
- RoCE v1 configuration is supported between Cisco adapters. Interoperability between Cisco adapters and third party adapters is not supported.
- Cisco UCS Manager does not support more than 4 RoCE-enabled vNICs per adapter.
- Cisco UCS Manager does not support RoCE with NVGRE, VXLAN, NetFlow, VMQ, or usNIC.
- After enabling RoCE v1 properties, enable no-drop QoS system class which is used in the vNIC QoS policy.
- Minimum number of queue pairs for the RoCE properties setting is 4.
- Maximum number of queue pairs per adapter is 8192.
- Maximum number of memory regions per adapter is 524288.
- If you do not disable RoCE before downgrading Cisco UCS Manager, downgrade will fail.
- Cisco UCS Manager does not support fabric failover for vNICs with RoCE enabled.

- A no-drop class QoS Policy is required when RoCE is enabled in the adapter policy for service profiles.

Guidelines for Using SMB Direct with RoCE v2

General Guidelines and Limitations:

- Cisco UCS Manager release 4.1.x and later releases support Microsoft SMB Direct with RoCE v2 on Microsoft Windows Server 2019. Cisco recommends that you have all KB updates from Microsoft for your Windows Server 2019.



Note RoCE v2 is not supported on Microsoft Windows Server 2016.

- Cisco recommends you check [UCS Hardware and Software Compatibility](#) specific to your UCS Manager release to determine support for Microsoft SMB Direct with RoCE v2 on Microsoft Windows 2019.
- Microsoft SMB Direct with RoCE v2 is supported only with fourth generation Cisco UCS VIC 1400 Series adapters. It is not supported with UCS VIC 12xx Series and 13xx Series adapters. SMB Direct with RoCE v2 is supported on all UCS Fabric Interconnects.



Note RoCE v1 is not supported with any fourth generation Cisco UCS VIC 1400 Series adapters.

- RoCE v2 configuration is supported only between Cisco adapters. Interoperability between Cisco adapters and third party adapters is not supported.
- RoCE v2 supports two RoCE v2 enabled vNIC per adapter and four virtual ports per adapter interface, independent of SET switch configuration.
- RoCE v2 cannot be used on the same vNIC interface as NVGRE, NetFlow, and VMQ features.
- RoCE v2 cannot be used with usNIC.
- RoCE v2-enabled vNIC interfaces must have the no-drop QoS system class enabled in UCS Manager.
- The RoCE Properties queue pairs setting must for be a minimum of 4 queue pairs.
- Maximum number of queue pairs per adapter is 2048.
- The QoS No Drop class configuration must be properly configured on upstream switches such as Cisco Nexus 9000 series switches. QoS configurations will vary between different upstream switches
- The maximum number of memory regions per rNIC interface is 131072.
- UCS Manager does not support fabric failover for vNICs with RoCE v2 enabled.

MTU Properties:

- In older versions of the VIC driver, the MTU was derived from either a UCS Manager service profile or from the Cisco IMC vNIC MTU setting in standalone mode. This behavior changed for 4th generation VIC 1400 Series adapters, where MTU is controlled from the Windows OS Jumbo Packet advanced property. A value configured from UCS Manager or Cisco IMC has no effect.

- The RoCE v2 MTU value is always power-of-two and its maximum limit is 4096.
- RoCE v2 MTU is derived from the Ethernet MTU.
- RoCE v2 MTU is the highest power-of-two that is less than the Ethernet MTU. For example:
 - if the Ethernet value is 1500, then the RoCE v2 MTU value is 1024
 - if the Ethernet value is 4096, then the RoCE v2 MTU value is 4096
 - if the Ethernet value is 9000, then the RoCE v2 MTU value is 4096

Windows NDPKI Modes of Operation:

- Cisco's implementation of Network Direct Kernel Provider Interface (NDPKI) supports two modes of operation: Mode 1 and Mode 2. Modes 1 and 2 relate to the implementation of Network Direct Kernel Provider Interface (NDKPI): Mode 1 is native RDMA, and Mode 2 involves configuration for the virtual port with RDMA. Cisco does not support NDPKI Mode 3 operation.
- The recommended default adapter policy for RoCE v2 Mode1 is Win-HPN-SMBd .
The recommended default adapter policy for RoCE v2 Mode2 is MQ-SMBd.
- RoCE v2 enabled vNICs for Mode2 operation require the QoS host control policy set to full.
- Mode 2 is inclusive of Mode 1: Mode 1 must be enabled to operate Mode 2.

Downgrade Limitations:

- Cisco recommends you remove the RoCE v2 configuration before downgrading to any non-supported RoCE v2 release. If the configuration is not removed or disabled, downgrade will fail.

Creating an Ethernet Adapter Policy



Tip

If the fields in an area do not display, click the **Expand** icon to the right of the heading.

SUMMARY STEPS

1. In the **Navigation** pane, click **Servers**.
2. Expand **Servers > Policies**.
3. Expand the node for the organization where you want to create the policy.
4. Right-click **Adapter Policies** and choose **Create Ethernet Adapter Policy**.
5. Enter a **Name** and optional **Description** for the policy.
6. (Optional) In the **Resources** area, adjust the following values:
7. (Optional) In the **Options** area, adjust the following values:
8. Click **OK**.
9. If a confirmation dialog box displays, click **Yes**.

DETAILED STEPS

Step 1 In the **Navigation** pane, click **Servers**.

Step 2 Expand **Servers > Policies**.

Step 3 Expand the node for the organization where you want to create the policy.

If the system does not include multitenancy, expand the **root** node.

Step 4 Right-click **Adapter Policies** and choose **Create Ethernet Adapter Policy**.

Step 5 Enter a **Name** and optional **Description** for the policy.

This name can be between 1 and 16 alphanumeric characters. You cannot use spaces or any special characters other than - (hyphen), _ (underscore), : (colon), and . (period), and you cannot change this name after the object is saved.

Step 6 (Optional) In the **Resources** area, adjust the following values:

Name	Description
Pooled radio button	Whether the queue resources are pooled or not. <ul style="list-style-type: none"> • Disabled—Pooling is disabled. • Enabled—Pooling is enabled. When pooling is enabled, the counts of queue resources specified in the Adapter Policy will be the total number of queues allocated across all vPorts.
Transmit Queues field	The number of transmit queue resources to allocate. Enter an integer between 1 and 1000.
Ring Size field	The number of descriptors in each transmit queue. Enter an integer between 64 and 4096.
Receive Queues field	The number of receive queue resources to allocate. Enter an integer between 1 and 1000.
Ring Size field	The number of descriptors in each receive queue. Enter an integer between 64 and 4096.
Completion Queues field	The number of completion queue resources to allocate. In general, the number of completion queue resources you should allocate is equal to the number of transmit queue resources plus the number of receive queue resources. Enter an integer between 1 and 2000.

Name	Description
Interrupts field	<p>The number of interrupt resources to allocate. In general, this value should be equal to (Completion Queues + 2) rounded up to nearest power of 2.</p> <p>Enter an integer between 1 and 1024.</p> <p>For example, if Transmit Queues = 1 and Receive Queues = 8 then:</p> <ul style="list-style-type: none"> • Completion Queues = 1 + 8 = 9 • Interrupt Count = (9 + 2) rounded up to the nearest power of 2 = 16

Step 7 (Optional) In the **Options** area, adjust the following values:

Name	Description
Transmit Checksum Offload radio button	<p>This can be one of the following:</p> <ul style="list-style-type: none"> • Disabled—The CPU calculates all packet checksums. • Enabled—The CPU sends all packets to the hardware so that the checksum can be calculated. This option may reduce CPU overhead. <p>Note This option affects only packets sent from the interface.</p>
Receive Checksum Offload radio button	<p>This can be one of the following:</p> <ul style="list-style-type: none"> • Disabled—The CPU validates all packet checksums. • Enabled—The CPU sends all packet checksums to the hardware for validation. This option may reduce CPU overhead. <p>Note This option affects only packets received by the interface.</p>
TCP Segmentation Offload radio button	<p>This can be one of the following:</p> <ul style="list-style-type: none"> • Disabled—The CPU segments large TCP packets. • Enabled—The CPU sends large TCP packets to the hardware to be segmented. This option may reduce CPU overhead and increase throughput rate. <p>Note This option is also known as Large Send Offload (LSO) and affects only packets sent from the interface.</p>
TCP Large Receive Offload radio button	<p>This can be one of the following:</p> <ul style="list-style-type: none"> • Disabled—The CPU processes all large packets. • Enabled—The hardware reassembles all segmented packets before sending them to the CPU. This option may reduce CPU utilization and increase inbound throughput. <p>Note This option affects only packets received by the interface.</p>

Name	Description
Receive Side Scaling radio button	<p>RSS distributes network receive processing across multiple CPUs in multiprocessor systems. This can be one of the following:</p> <ul style="list-style-type: none"> • Disabled—Network receive processing is always handled by a single processor even if additional processors are available. • Enabled—Network receive processing is shared across processors whenever possible.
Accelerated Receive Flow Steering radio button	<p>Packet processing for a flow must be performed on the local CPU. This is supported for Linux operating systems only. This can be one of the following:</p> <ul style="list-style-type: none"> • Disabled—The CPU is not specified. • Enabled—Packet processing is performed on the local CPU.
Network Virtualization using Generic Routing Encapsulation radio button	<p>Whether NVGRE overlay hardware offloads for TSO and checksum are enabled. This can be one of the following:</p> <ul style="list-style-type: none"> • Disabled—NVGRE overlay hardware offloads are not enabled. • Enabled—NVGRE overlay hardware offloads are enabled. <p>NVGRE overlay hardware offloads can be enabled when using UCS VIC 14xx adapters.</p>
Virtual Extensible LAN radio button	<p>Whether VXLAN overlay hardware offloads for TSO and checksum are enabled. This can be one of the following:</p> <ul style="list-style-type: none"> • Disabled—VXLAN overlay hardware offloads are not enabled. • Enabled—VXLAN overlay hardware offloads are enabled. <p>VXLAN overlay hardware offloads can be enabled with RoCE and VMQ when using UCS VIC 14xx adapters.</p>
Failback Timeout field	<p>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.</p> <p>Enter a number of seconds between 0 and 600.</p>
Interrupt Mode radio button	<p>The preferred driver interrupt mode. This can be one of the following:</p> <ul style="list-style-type: none"> • MSI X—Message Signaled Interrupts (MSI) with the optional extension. This is the recommended option. <p>Note If you set Interrupt Mode as Msi-X, and if pci=noms parameter is enabled in <code>/boot/grub/grub.conf</code> on RHEL system, then pci=noms would block the eNIC/fNIC driver to run in the Msi-X mode, impacting system performance.</p> <ul style="list-style-type: none"> • MSI—MSI only. • IN Tx—PCI IN Tx interrupts.

Name	Description
Interrupt Coalescing Type radio button	<p>This can be one of the following:</p> <ul style="list-style-type: none"> • Min—The system waits for the time specified in the Interrupt Timer 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 Interrupt Timer field.
Interrupt Timer field	<p>The time to wait between interrupts or the idle period that must be encountered before an interrupt is sent.</p> <p>Enter a value between 1 and 65535. To turn off interrupt coalescing, enter 0 (zero) in this field.</p>
RoCE radio button	<p>Whether Remote Direct Memory Access over an Ethernet network is enabled. This can be one of the following:</p> <ul style="list-style-type: none"> • Disabled—RoCE is disabled on the Ethernet adapter. • Enabled—RoCE is enabled on the Ethernet adapter.
RoCE Properties area	Lists the RoCE properties. This area is enabled only if you enable RoCE.
Version 1 radio button	<p>RoCE Version 1 is a link layer protocol. It allows communication between any two hosts in the same Ethernet broadcast domain.</p> <p>Whether RoCE Version 1 is enabled. This can be one of the following:</p> <ul style="list-style-type: none"> • Disabled—RoCE version 1 is disabled on the Ethernet adapter. • Enabled—RoCE version 1 is enabled on the Ethernet adapter.
Version 2 radio button	<p>For Future Enablement:</p> <p>RoCEv2 is an internet layer protocol. RoCEv2 packets can be routed. This is possible because RoCEv2 packets now include an IP and UDP header.</p> <p>Whether RoCE Version 2 is enabled. This can be one of the following:</p> <ul style="list-style-type: none"> • Disabled—RoCE version 2 is disabled on the Ethernet adapter. • Enabled—RoCE version 2 is enabled on the Ethernet adapter. <p>If you enable RoCE version 2, you can also set the Priority field.</p>
Queue Pairs field	<p>The number of queue pairs per adapter.</p> <p>Enter an integer between 1 and 8192. It is recommended that this number be an integer power of 2.</p>

Name	Description
Priority drop-down list	<p>Pre-defined set of Global (system wide) QoS classes. These are:</p> <ul style="list-style-type: none"> • Fibre Channel • Best Effort • Bronze • Silver • Gold • Platinum <p>For RoCE version 2, set Priority as Platinum.</p>
Memory Regions field	<p>The number of memory regions per adapter.</p> <p>Enter an integer between 1 and 524288. It is recommended that this number be an integer power of 2.</p>
Resource Groups field	<p>The number of resource groups per adapter.</p> <p>Enter an integer between 1 and 128.</p> <p>It is recommended that this number be an integer power of 2 greater than or equal to the number of CPU cores on the system for optimum performance.</p>
Advance Filter radio button	<p>Whether Advance Filter over an Ethernet network is enabled. This can be one of the following:</p> <ul style="list-style-type: none"> • Disabled—Advance filter is disabled on the Ethernet adapter. • Enabled—Advance filter is enabled on the Ethernet adapter.
Interrupt Scaling radio button	<p>Whether Interrupt Scaling over an Ethernet network is enabled. This can be one of the following:</p> <ul style="list-style-type: none"> • Disabled—Interrupt Scaling is disabled on the Ethernet adapter. • Enabled—Interrupt Scaling is enabled on the Ethernet adapter.

Step 8 Click **OK**.

Step 9 If a confirmation dialog box displays, click **Yes**.

Configuring an Ethernet Adapter Policy to Enable eNIC Support for MRQS on Linux Operating Systems

Cisco UCS Manager includes eNIC support for the Multiple Receive Queue Support (MRQS) feature on Red Hat Enterprise Linux Version 6.x and SUSE Linux Enterprise Server Version 11.x.

SUMMARY STEPS

1. Create an Ethernet adapter policy.
2. Install an eNIC driver Version 2.1.1.35 or later.
3. Reboot the server.

DETAILED STEPS

Step 1 Create an Ethernet adapter policy.

Use the following parameters when creating the Ethernet adapter policy:

- Transmit Queues = 1
- Receive Queues = n (up to 8)
- Completion Queues = # of Transmit Queues + # of Receive Queues
- Interrupts = # Completion Queues + 2
- Receive Side Scaling (RSS) = Enabled
- Interrupt Mode = Msi-X

Note If you set **Interrupt Mode** as **Msi-X**, and if **pci=noms** parameter is enabled in `/boot/grub/grub.conf` on RHEL system, then **pci=noms** would block the eNIC/fNIC driver to run in the **Msi-X** mode, impacting system performance.

Step 2 Install an eNIC driver Version 2.1.1.35 or later.

For more information, see the *Cisco UCS Virtual Interface Card Drivers Installation Guide*.

Step 3 Reboot the server.

Configuring an Ethernet Adapter Policy to Enable eNIC Support for RSS on VMware ESXi

Cisco UCS Manager includes eNIC support for the Receive Side Scaling (RSS) feature on ESXi 5.5 and later releases.

SUMMARY STEPS

1. Create an Ethernet adapter policy.
2. Install the appropriate drivers according to the [UCS Hardware and Software Compatibility](#).
3. Reboot the server.

DETAILED STEPS

Step 1 Create an Ethernet adapter policy.

Use the following parameters when creating the Ethernet adapter policy.

In the **Resources** area, set the following options:

- Transmit Queues = 1
- Receive Queues = n (up to 16)
- Completion Queues = # of Transmit Queues + # of Receive Queues
- Interrupts = # Completion Queues + 2

In the **Options** area, set the following option:

- Receive Side Scaling (RSS) = Enabled

- Step 2** Install the appropriate drivers according to the [UCS Hardware and Software Compatibility](#).
For more information, see the *Cisco UCS Virtual Interface Card Drivers Installation Guide*.
- Step 3** Reboot the server.

Configuring an Ethernet Adapter Policy to Enable Stateless Offloads with NVGRE

Cisco UCS Manager supports stateless offloads with NVGRE on Cisco UCS VIC 13XX adapters that are installed on servers running Windows Server 2012 R2 operating systems and higher versions. NVGRE feature is also supported on servers with Cisco UCS VIC 14XX running Windows Server 2016. Stateless offloads with NVGRE cannot be used with Netflow, usNIC, or VM-FEX.

SUMMARY STEPS

1. In the **Navigation** pane, click **Servers**.
2. Expand **Servers > Policies**.
3. Expand the node for the organization where you want to create the policy.
4. Right-click **Adapter Policies** and choose **Create Ethernet Adapter Policy**.
5. Click **OK** to create the Ethernet adapter policy.
6. Install an eNIC driver Version 3.0.0.8 or later.
7. Reboot the server.

DETAILED STEPS

-
- Step 1** In the **Navigation** pane, click **Servers**.
- Step 2** Expand **Servers > Policies**.
- Step 3** Expand the node for the organization where you want to create the policy.
If the system does not include multitenancy, expand the **root** node.
- Step 4** Right-click **Adapter Policies** and choose **Create Ethernet Adapter Policy**.
a) In the **Resources** area, set the following options:

- Transmit Queues = 1
- Receive Queues = n (up to 8)
- Completion Queues = # of Transmit Queues + # of Receive Queues
- Interrupts = # Completion Queues + 2

b) In the **Options** area, set the following options:

- Network Virtualization using Generic Routing Encapsulation = Enabled
- Interrupt Mode = Msi-X

Note If you set **Interrupt Mode** as **Msi-X**, and if **pci=noms**i parameter is enabled in `/boot/grub/grub.conf` on RHEL system, then **pci=noms**i would block the eNIC/fNIC driver to run in the **Msi-X** mode, impacting system performance.

For more information on creating an Ethernet adapter policy, see [Creating an Ethernet Adapter Policy, on page 188](#).

Step 5 Click **OK** to create the Ethernet adapter policy.

Step 6 Install an eNIC driver Version 3.0.0.8 or later.

For more information, see the *Cisco UCS Virtual Interface Card Drivers Installation Guide*.

Step 7 Reboot the server.

Configuring an Ethernet Adapter Policy to Enable Stateless Offloads with VXLAN

Cisco UCS Manager supports VXLAN TSO and checksum offloads only with Cisco UCS VIC 13XX adapters that are running on ESXi 5.5 and later releases.

VXLAN with Receive Side-Scaling (RSS) support starts with the Cisco UCS Manager 3.1(2) release. RSS is supported with VXLAN stateless offload on VIC adapters 13XX and SIOC on Cisco UCS S3260 system for ESXi 5.5 and later releases.

Cisco UCS Manager 4.0(1a) Release introduces VXLAN support on servers with Cisco UCS VIC 14XX running ESXi 6.5 and later releases. Stateless offloads with VXLAN cannot be used with NetFlow, usNIC, VM-FEX, or Netqueue.

VXLAN support for Linux and Windows 2016 starts with Cisco UCS Manager 4.0(1a) for VIC 14XX adapters. The maximum amount of receive queues may be up to 16 for VIC 13XX and 14XX adapters on ESXi.



Note

VXLAN stateless hardware offloads are not supported with Guest OS TCP traffic over IPv6 on UCS VIC 13xx adapters. To run VXLAN encapsulated TCP traffic over IPV6, disable the VXLAN stateless offloads feature.

- To disable the VXLAN stateless offload feature in UCS Manager, disable 'Virtual Extensible LAN' field in the Ethernet Adapter Policy.

SUMMARY STEPS

1. In the **Navigation** pane, click **Servers**.
2. Expand **Servers > Policies**.
3. Expand the node for the organization where you want to create the policy.
4. Right-click **Adapter Policies** and choose **Create Ethernet Adapter Policy**.
5. Click **OK** to create the Ethernet adapter policy.
6. Install an eNIC driver Version 2.1.2.59 or later.
7. Reboot the server.

DETAILED STEPS

Step 1 In the **Navigation** pane, click **Servers**.

Step 2 Expand **Servers > Policies**.

Step 3 Expand the node for the organization where you want to create the policy.

If the system does not include multitenancy, expand the **root** node.

Step 4 Right-click **Adapter Policies** and choose **Create Ethernet Adapter Policy**.

a) In the **Resources** area, set the following options:

- Transmit Queues = 1
- Receive Queues = n (up to 16)
- Completion Queues = # of Transmit Queues + # of Receive Queues
- Interrupts = # Completion Queues + 2

b) In the **Options** area, set the following options:

- Receive Side Scaling = Enabled
- Virtual Extensible LAN = Enabled
- Interrupt Mode = Msi-X

Note If you set **Interrupt Mode** as **Msi-X**, and if **pci=noms**i parameter is enabled in `/boot/grub/grub.conf` on RHEL system, then **pci=noms**i would block the eNIC/fNIC driver to run in the **Msi-X** mode, impacting system performance.

For more information on creating an ethernet adapter policy, see [Creating an Ethernet Adapter Policy, on page 188](#).

Step 5 Click **OK** to create the Ethernet adapter policy.

Step 6 Install an eNIC driver Version 2.1.2.59 or later.

For more information, see the *Cisco UCS Virtual Interface Card Drivers Installation Guide*.

Step 7 Reboot the server.

Deleting an Ethernet Adapter Policy

SUMMARY STEPS

1. In the **Navigation** pane, click **LAN**.
2. Expand **LAN > Policies > *Organization_Name***.
3. Expand the **Adapter Policies** node.
4. Right-click the Ethernet adapter policy that you want to delete and choose **Delete**.
5. If a confirmation dialog box displays, click **Yes**.

DETAILED STEPS

-
- | | |
|---------------|--|
| Step 1 | In the Navigation pane, click LAN . |
| Step 2 | Expand LAN > Policies > <i>Organization_Name</i> . |
| Step 3 | Expand the Adapter Policies node. |
| Step 4 | Right-click the Ethernet adapter policy that you want to delete and choose Delete . |
| Step 5 | If a confirmation dialog box displays, click Yes . |
-

Configuring the Default vNIC Behavior Policy

Default vNIC Behavior Policy

Default vNIC behavior policy allows you to configure how vNICs are created for a service profile. You can choose to create vNICs manually, or you can create them automatically.

You can configure the default vNIC behavior policy to define how vNICs are created. This can be one of the following:

- **None**—Cisco UCS Manager does not create default vNICs for a service profile. All vNICs must be explicitly created.
- **HW Inherit**—If a service profile requires vNICs and none have been explicitly defined, Cisco UCS Manager creates the required vNICs based on the adapter installed in the server associated with the service profile.

**Note**

If you do not specify a default behavior policy for vNICs, **HW Inherit** is used by default.

Configuring a Default vNIC Behavior Policy

SUMMARY STEPS

1. In the **Navigation** pane, click **LAN**.
2. Expand **LAN > Policies**.
3. Expand the **root** node.
4. Click **Default vNIC Behavior**.
5. On the **General Tab**, in the **Properties** area, click one of the following radio buttons in the **Action** field:
6. Click **Save Changes**.

DETAILED STEPS

Step 1 In the **Navigation** pane, click **LAN**.

Step 2 Expand **LAN > Policies**.

Step 3 Expand the **root** node.

You can configure only the default vNIC behavior policy in the root organization. You cannot configure the default vNIC behavior policy in a sub-organization.

Step 4 Click **Default vNIC Behavior**.

Step 5 On the **General Tab**, in the **Properties** area, click one of the following radio buttons in the **Action** field:

- **None**—Cisco UCS Manager does not create default vNICs for a service profile. All vNICs must be explicitly created.
- **HW Inherit**—If a service profile requires vNICs and none have been explicitly defined, Cisco UCS Manager creates the required vNICs based on the adapter installed in the server associated with the service profile.

Step 6 Click **Save Changes**.

Configuring LAN Connectivity Policies

About the LAN and SAN Connectivity Policies

Connectivity policies determine the connections and the network communication resources between the server and the LAN or SAN on the network. These policies use pools to assign MAC addresses, WWNs, and WWPNs to servers and to identify the vNICs and vHBAs that the servers use to communicate with the network.

**Note**

We do not recommend that you use static IDs in connectivity policies, because these policies are included in service profiles and service profile templates and can be used to configure multiple servers.

Privileges Required for LAN and SAN Connectivity Policies

Connectivity policies enable users without network or storage privileges to create and modify service profiles and service profile templates with network and storage connections. However, users must have the appropriate network and storage privileges to create connectivity policies.

Privileges Required to Create Connectivity Policies

Connectivity policies require the same privileges as other network and storage configurations. For example, you must have at least one of the following privileges to create connectivity policies:

- **admin**—Can create LAN and SAN connectivity policies
- **ls-server**—Can create LAN and SAN connectivity policies
- **ls-network**—Can create LAN connectivity policies
- **ls-storage**—Can create SAN connectivity policies

Privileges Required to Add Connectivity Policies to Service Profiles

After the connectivity policies have been created, a user with **ls-compute** privileges can include them in a service profile or service profile template. However, a user with only **ls-compute** privileges cannot create connectivity policies.

Interactions between Service Profiles and Connectivity Policies

You can configure the LAN and SAN connectivity for a service profile through either of the following methods:

- LAN and SAN connectivity policies that are referenced in the service profile
- Local vNICs and vHBAs that are created in the service profile
- Local vNICs and a SAN connectivity policy
- Local vHBAs and a LAN connectivity policy

Cisco UCS maintains mutual exclusivity between connectivity policies and local vNIC and vHBA configuration in the service profile. You cannot have a combination of connectivity policies and locally created vNICs or vHBAs. When you include a LAN connectivity policy in a service profile, all existing vNIC configuration is erased, and when you include a SAN connectivity policy, all existing vHBA configuration in that service profile is erased.

Creating a LAN Connectivity Policy

SUMMARY STEPS

1. In the **Navigation** pane, click **LAN**.
2. Expand **LAN > Policies**.
3. Expand the node for the organization where you want to create the policy.
4. Right-click **LAN Connectivity Policies** and choose **Create LAN Connectivity Policy**.
5. In the **Create LAN Connectivity Policy** dialog box, enter a name and optional description.

6. Do one of the following:
 - To add vNICs to the LAN connectivity policy, continue with Step 7.
 - To add iSCSI vNICs to the LAN connectivity policy and use iSCSI boot with the server, continue with Step 8.
7. To add vNICs, click **Add** next to the plus sign and complete the following fields in the **Create vNIC** dialog box:
8. If you want to use iSCSI boot with the server, click the down arrows to expand the **Add iSCSI vNICs** bar and do the following:
9. After you have created all the vNICs or iSCSI vNICs you need for the policy, click **OK**.

DETAILED STEPS

-
- Step 1** In the **Navigation** pane, click **LAN**.
- Step 2** Expand **LAN > Policies**.
- Step 3** Expand the node for the organization where you want to create the policy.
If the system does not include multitenancy, expand the **root** node.
- Step 4** Right-click **LAN Connectivity Policies** and choose **Create LAN Connectivity Policy**.
- Step 5** In the **Create LAN Connectivity Policy** dialog box, enter a name and optional description.
- Step 6** Do one of the following:
- To add vNICs to the LAN connectivity policy, continue with Step 7.
 - To add iSCSI vNICs to the LAN connectivity policy and use iSCSI boot with the server, continue with Step 8.
- Step 7** To add vNICs, click **Add** next to the plus sign and complete the following fields in the **Create vNIC** dialog box:
- a) In the **Create vNIC** dialog box, enter the name, select a **MAC Address Assignment**, and check the **Use vNIC Template** check box to use an existing vNIC template.
You can also create a MAC pool from this area.
 - b) Choose the **Fabric ID**, select the **VLANs** that you want to use, enter the **MTU**, and choose a **Pin Group**.
You can also create a VLAN and a LAN pin group from this area.
- Note** Cisco recommends using the native VLAN 1 setting to prevent traffic interruptions if using the Cisco Nexus 1000V Series Switches because changing the native VLAN 1 setting on a vNIC causes the port to turn on and off. You can only change the native VLAN setting on a Virtual Private Cloud (VPC) secondary port, and then change the primary port on the VPC.
- c) In the **Operational Parameters** area, choose a **Stats Threshold Policy**.
 - d) In the Adapter Performance Profile area, choose an **Adapter Policy**, **QoS Policy**, and a **Network Control Policy**.
You can also create an Ethernet adapter policy, QoS policy, and network control policy from this area.
 - e) In the Connection Policies area, choose the **Dynamic vNIC**, **usNIC** or **VMQ** radio button, then choose the corresponding policy.
You can also create a dynamic vNIC, usNIC, or VMQ connection policy from this area.
- Note** Cisco UCS 6400 Series Fabric Interconnects do not support dynamic vNICs.

f) Click **OK**.

Step 8

If you want to use iSCSI boot with the server, click the down arrows to expand the **Add iSCSI vNICs** bar and do the following:

- a) Click **Add** on the table icon bar.
- b) In the **Create iSCSI vNIC** dialog box, enter the **Name** and choose the **Overlay vNIC**, **iSCSI Adapter Policy**, and **VLAN**.

You can also create an iSCSI adapter policy from this area.

Note For the Cisco UCS M81KR Virtual Interface Card and the Cisco UCS VIC-1240 Virtual Interface Card, the VLAN that you specify must be the same as the native VLAN on the overlay vNIC.

For the Cisco UCS M51KR-B Broadcom BCM57711 Adapter, the VLAN that you specify can be any VLAN assigned to the overlay vNIC.

- c) In the **MAC Address Assignment** drop-down list in the **iSCSI MAC Address** area, choose one of the following:
 - Leave the MAC address unassigned, select **Select (None used by default)**. Select this option if the server that will be associated with this service profile contains a Cisco UCS M81KR Virtual Interface Card adapter or a Cisco UCS VIC-1240 Virtual Interface Card.
 - **Important** If the server that will be associated with this service profile contains a Cisco UCS NIC M51KR-B adapter, you must specify a MAC address.
 - A specific MAC address, select **00:25:B5:XX:XX:XX** and enter the address in the **MAC Address** field. To verify that this address is available, click the corresponding link.
 - A MAC address from a pool, select the pool name from the list. Each pool name is followed by a pair of numbers in parentheses. The first number is the number of available MAC addresses in the pool and the second is the total number of MAC addresses in the pool.

If this Cisco UCS domain is registered with Cisco UCS Central, there might be two pool categories. **Domain Pools** are defined locally in the Cisco UCS domain and **Global Pools** are defined in Cisco UCS Central.

- d) (Optional) If you want to create a MAC pool that will be available to all service profiles, click **Create MAC Pool** and complete the fields in the **Create MAC Pool** wizard.

For more information, see the *UCS Manager Storage Management Guide*, Pools chapter, Creating a MAC Pool topic.

e) Click **OK**.

Step 9

After you have created all the vNICs or iSCSI vNICs you need for the policy, click **OK**.

What to do next

Include the policy in a service profile or service profile template.

Deleting a LAN Connectivity Policy

If you delete a LAN connectivity policy that is included in a service profile, it also deletes all vNICs and iSCSI vNICs from that service profile, and disrupt LAN data traffic for the server associated with the service profile.

SUMMARY STEPS

1. In the **Navigation** pane, click **LAN**.
2. Expand **LAN > Policies > *Organization_Name***.
3. Expand the **LAN Connectivity Policies** node.
4. Right-click the policy that you want to delete and choose **Delete**.
5. If a confirmation dialog box displays, click **Yes**.

DETAILED STEPS

-
- | | |
|---------------|---|
| Step 1 | In the Navigation pane, click LAN . |
| Step 2 | Expand LAN > Policies > <i>Organization_Name</i> . |
| Step 3 | Expand the LAN Connectivity Policies node. |
| Step 4 | Right-click the policy that you want to delete and choose Delete . |
| Step 5 | If a confirmation dialog box displays, click Yes . |
-

Creating a vNIC for a LAN Connectivity Policy

SUMMARY STEPS

1. In the **Navigation** pane, click **LAN**.
2. Expand **LAN > Policies > *Organization_Name***.
3. Expand the **LAN Connectivity Policies** node.
4. Choose the policy to which you want to add a vNIC.
5. In the **Work** pane, click the **General** tab.
6. On the icon bar of the vNICs table, click **Add**.
7. In the **Create vNIC** dialog box, enter the name, select a **MAC Address Assignment**, and check the **Use vNIC Template** check box if you want to use an existing vNIC template.
8. Choose the **Fabric ID**, select the **VLANS** that you want to use, enter the **MTU**, and choose a **Pin Group**.
9. In the **Operational Parameters** area, choose a **Stats Threshold Policy**.
10. In the Adapter Performance Profile area, choose an **Adapter Policy**, **QoS Policy**, and a **Network Control Policy**.
11. In the Connection Policies area, choose the **Dynamic vNIC**, **usNIC** or **VMQ** radio button, then choose the corresponding policy.
12. Click **OK**.
13. Click **Save Changes**.

DETAILED STEPS

-
- | | |
|---------------|---|
| Step 1 | In the Navigation pane, click LAN . |
| Step 2 | Expand LAN > Policies > <i>Organization_Name</i> . |
| Step 3 | Expand the LAN Connectivity Policies node. |
| Step 4 | Choose the policy to which you want to add a vNIC. |

- Step 5** In the **Work** pane, click the **General** tab.
- Step 6** On the icon bar of the vNICs table, click **Add**.
- Step 7** In the **Create vNIC** dialog box, enter the name, select a **MAC Address Assignment**, and check the **Use vNIC Template** check box if you want to use an existing vNIC template.
- You can also create a MAC pool from this area.
- Step 8** Choose the **Fabric ID**, select the **VLANs** that you want to use, enter the **MTU**, and choose a **Pin Group**.
- You can also create a VLAN and a LAN pin group from this area.
- Step 9** In the **Operational Parameters** area, choose a **Stats Threshold Policy**.
- Step 10** In the Adapter Performance Profile area, choose an **Adapter Policy**, **QoS Policy**, and a **Network Control Policy**.
- You can also create an Ethernet adapter policy, QoS policy, and network control policy from this area.
- Step 11** In the Connection Policies area, choose the **Dynamic vNIC**, **usNIC** or **VMQ** radio button, then choose the corresponding policy.
- You can also create a dynamic vNIC, usNIC, or VMQ connection policy from this area.
- Note** Cisco UCS 6400 Series Fabric Interconnects do not support dynamic vNICs.
- Step 12** Click **OK**.
- Step 13** Click **Save Changes**.

Deleting a vNIC from a LAN Connectivity Policy

SUMMARY STEPS

1. In the **Navigation** pane, click **LAN**.
2. Expand **LAN > Policies > Organization_Name**.
3. Expand the **LAN Connectivity Policies** node.
4. Select the policy from which you want to delete the vNIC.
5. In the **Work** pane, click the **General** tab.
6. In the vNICs table, do the following:
7. If a confirmation dialog box displays, click **Yes**.
8. Click **Save Changes**.

DETAILED STEPS

- Step 1** In the **Navigation** pane, click **LAN**.
- Step 2** Expand **LAN > Policies > Organization_Name**.
- Step 3** Expand the **LAN Connectivity Policies** node.
- Step 4** Select the policy from which you want to delete the vNIC.
- Step 5** In the **Work** pane, click the **General** tab.
- Step 6** In the vNICs table, do the following:

- a) Click the vNIC you want to delete.
- b) On the icon bar, click **Delete**.

Step 7 If a confirmation dialog box displays, click **Yes**.

Step 8 Click **Save Changes**.

Creating an iSCSI vNIC for a LAN Connectivity Policy

SUMMARY STEPS

1. In the **Navigation** pane, click **LAN**.
2. Expand **LAN > Policies > *Organization_Name***.
3. Expand the **LAN Connectivity Policies** node.
4. Choose the policy to which you want to add an iSCSI vNIC.
5. In the **Work** pane, click the **General** tab.
6. On the icon bar of the **Add iSCSI vNICs** table, click **Add**.
7. In the **Create iSCSI vNIC** dialog box, complete the following fields:
8. In the **MAC Address Assignment** drop-down list in the **iSCSI MAC Address** area, choose one of the following:
9. (Optional) If you want to create a MAC pool that will be available to all service profiles, click **Create MAC Pool** and complete the fields in the **Create MAC Pool** wizard.
10. Click **OK**.
11. Click **Save Changes**.

DETAILED STEPS

- Step 1** In the **Navigation** pane, click **LAN**.
- Step 2** Expand **LAN > Policies > *Organization_Name***.
- Step 3** Expand the **LAN Connectivity Policies** node.
- Step 4** Choose the policy to which you want to add an iSCSI vNIC.
- Step 5** In the **Work** pane, click the **General** tab.
- Step 6** On the icon bar of the **Add iSCSI vNICs** table, click **Add**.
- Step 7** In the **Create iSCSI vNIC** dialog box, complete the following fields:

Name	Description
Name field	The name of the iSCSI vNIC. This name can be between 1 and 16 alphanumeric characters. You cannot use spaces or any special characters other than - (hyphen), _ (underscore), : (colon), and . (period), and you cannot change this name after the object is saved.
Overlay vNIC drop-down list	The LAN vNIC associated with this iSCSI vNIC, if any.
iSCSI Adapter Policy drop-down list	The iSCSI adapter policy associated with this iSCSI vNIC, if any.

Name	Description
Create iSCSI Adapter Policy link	Click this link to create a new iSCSI adapter policy that will be available to all iSCSI vNICs.
VLAN drop-down list	<p>The virtual LAN associated with this iSCSI vNIC. The default VLAN is default.</p> <p>Note For the Cisco UCS M81KR Virtual Interface Card and the Cisco UCS VIC-1240 Virtual Interface Card, the VLAN that you specify must be the same as the native VLAN on the overlay vNIC.</p> <p>For the Cisco UCS M51KR-B Broadcom BCM57711 Adapter, the VLAN that you specify can be any VLAN assigned to the overlay vNIC.</p>

Step 8

In the **MAC Address Assignment** drop-down list in the **iSCSI MAC Address** area, choose one of the following:

- Leave the MAC address unassigned, select **Select (None used by default)**. Select this option if the server that will be associated with this service profile contains a Cisco UCS M81KR Virtual Interface Card adapter or a Cisco UCS VIC-1240 Virtual Interface Card.
- **Important** If the server that will be associated with this service profile contains a Cisco UCS NIC M51KR-B adapter, you must specify a MAC address.
- A specific MAC address, select **00:25:B5:XX:XX:XX** and enter the address in the **MAC Address** field. To verify that this address is available, click the corresponding link.
- A MAC address from a pool, select the pool name from the list. Each pool name is followed by a pair of numbers in parentheses. The first number is the number of available MAC addresses in the pool and the second is the total number of MAC addresses in the pool.

If this Cisco UCS domain is registered with Cisco UCS Central, there might be two pool categories. **Domain Pools** are defined locally in the Cisco UCS domain and **Global Pools** are defined in Cisco UCS Central.

Step 9

(Optional) If you want to create a MAC pool that will be available to all service profiles, click **Create MAC Pool** and complete the fields in the **Create MAC Pool** wizard.

For more information, see the *UCS Manager Storage Management Guide*, Pools chapter, Creating a MAC Pool topic.

Step 10

Click **OK**.

Step 11

Click **Save Changes**.

Deleting a vNIC from a LAN Connectivity Policy

SUMMARY STEPS

1. In the **Navigation** pane, click **LAN**.
2. Expand **LAN > Policies > Organization_Name**.
3. Expand the **LAN Connectivity Policies** node.
4. Select the policy from which you want to delete the vNIC.

5. In the **Work** pane, click the **General** tab.
6. In the **vNICs** table, do the following:
7. If a confirmation dialog box displays, click **Yes**.
8. Click **Save Changes**.

DETAILED STEPS

-
- | | |
|---------------|--|
| Step 1 | In the Navigation pane, click LAN . |
| Step 2 | Expand LAN > Policies > <i>Organization_Name</i> . |
| Step 3 | Expand the LAN Connectivity Policies node. |
| Step 4 | Select the policy from which you want to delete the vNIC. |
| Step 5 | In the Work pane, click the General tab. |
| Step 6 | In the vNICs table, do the following:
a) Click the vNIC you want to delete.
b) On the icon bar, click Delete . |
| Step 7 | If a confirmation dialog box displays, click Yes . |
| Step 8 | Click Save Changes . |
-

Configuring Network Control Policies

Network Control Policy

This policy configures the network control settings for the Cisco UCS domain, including the following:

- Whether the Cisco Discovery Protocol (CDP) is enabled or disabled
- How the virtual interface (VIF) behaves if no uplink port is available in end-host mode
- The action that Cisco UCS Manager takes on the remote Ethernet interface, vEthernet interface , or vFibre Channel interface when the associated border port fails
- Whether the server can use different MAC addresses when sending packets to the fabric interconnect
- Whether MAC registration occurs on a per-VNIC basis or for all VLANs

Action on Uplink Fail

By default, the **Action on Uplink Fail** property in the network control policy is configured with a value of link-down. For adapters such as the Cisco UCS M81KR Virtual Interface Card, this default behavior directs Cisco UCS Manager to bring the vEthernet or vFibre Channel interface down if the associated border port fails. For Cisco UCS systems using a non-VM-FEX capable converged network adapter that supports both Ethernet and FCoE traffic, such as Cisco UCS CNA M72KR-Q and the Cisco UCS CNA M72KR-E, this default behavior directs Cisco UCS Manager to bring the remote Ethernet interface down if the associated border port fails. In this scenario, any vFibre Channel interfaces that are bound to the remote Ethernet interface are brought down as well.



Note If your implementation includes those types of non-VM-FEX capable converged network adapters mentioned in this section and the adapter is expected to handle both Ethernet and FCoE traffic, we recommend that you configure the **Action on Uplink Fail** property with a value of warning. Note that this configuration might result in an Ethernet teaming driver not being able to detect a link failure when the border port goes down.

MAC Registration Mode

MAC addresses are installed only on the native VLAN by default, which maximizes the VLAN port count in most implementations.



Note If a trunking driver is being run on the host and the interface is in promiscuous mode, we recommend that you set the MAC Registration Mode to All VLANs.

NIC Teaming and Port Security

NIC teaming is a grouping together of network adapters to build in redundancy, and is enabled on the host. This teaming or bonding facilitates various functionalities, including load balancing across links and failover. When NIC teaming is enabled and events such as failover or reconfiguration take place, MAC address conflicts and movement may happen.

Port security, which is enabled on the fabric interconnect side, prevents MAC address movement and deletion. Therefore, you must not enable port security and NIC teaming together.

Configuring Link Layer Discovery Protocol for Fabric Interconnect vEthernet Interfaces

Cisco UCS Manager allows you to enable and disable LLDP on a vEthernet interface. You can also retrieve information about these LAN uplink neighbors. This information is useful while learning the topology of the LAN connected to the UCS system and while diagnosing any network connectivity issues from the fabric interconnect (FI). The fabric interconnect of a UCS system is connected to LAN uplink switches for LAN connectivity and to SAN uplink switches for storage connectivity. When using Cisco UCS with Cisco Application Centric Infrastructure (ACI), LAN uplinks of the fabric interconnect are connected to ACI leaf nodes. Enabling LLDP on a vEthernet interface will help the Application Policy Infrastructure Controller (APIC) to identify the servers connected to the fabric interconnect by using vCenter.

To permit the discovery of devices in a network, support for Link Layer Discovery Protocol (LLDP), a vendor-neutral device discovery protocol that is defined in the IEEE 802.1ab standard, is introduced. LLDP is a one-way protocol that allows network devices to advertise information about themselves to other devices on the network. LLDP transmits information about the capabilities and current status of a device and its interfaces. LLDP devices use the protocol to solicit information only from other LLDP devices.

You can enable or disable LLDP on a vEthernet interface based on the Network Control Policy (NCP) that is applied on the vNIC in the service profile.

Creating a Network Control Policy

MAC address-based port security for Emulex converged Network Adapters (N20-AE0102) is not supported. When MAC address-based port security is enabled, the fabric interconnect restricts traffic to packets that contain the MAC address that it first learns. This is either the source MAC address used in the FCoE

Initialization Protocol packet, or the MAC address in an ethernet packet, whichever is sent first by the adaptor. This configuration can result in either FCoE or Ethernet packets being dropped.

SUMMARY STEPS

1. In the **Navigation** pane, click **LAN**.
2. Expand **LAN > Policies**.
3. Expand the node for the organization where you want to create the policy.
4. Right-click the **Network Control Policies** node and select **Create Network Control Policy**.
5. In the **Create Network Control Policy** dialog box, complete the required fields.
6. In the **LLDP** area, do the following:
7. In the **MAC Security** area, do the following to determine whether the server can use different MAC addresses when sending packets to the fabric interconnect:
8. Click **OK**.

DETAILED STEPS

-
- | | |
|---------------|---|
| Step 1 | In the Navigation pane, click LAN . |
| Step 2 | Expand LAN > Policies . |
| Step 3 | Expand the node for the organization where you want to create the policy.

If the system does not include multitenancy, expand the root node. |
| Step 4 | Right-click the Network Control Policies node and select Create Network Control Policy . |
| Step 5 | In the Create Network Control Policy dialog box, complete the required fields. |
| Step 6 | In the LLDP area, do the following:
<ol style="list-style-type: none">a) To enable the transmission of LLDP packets on an interface, click Enabled in the Transmit field.b) To enable the reception of LLDP packets on an interface, click Enabled in the Receive field. |
| Step 7 | In the MAC Security area, do the following to determine whether the server can use different MAC addresses when sending packets to the fabric interconnect:
<ol style="list-style-type: none">a) Click the Expand icon to expand the area and display the radio buttons.b) Click one of the following radio buttons to determine whether forged MAC addresses are allowed or denied when packets are sent from the server to the fabric interconnect:<ul style="list-style-type: none">• Allow— All server packets are accepted by the fabric interconnect, regardless of the MAC address associated with the packets.• Deny— After the first packet has been sent to the fabric interconnect, all other packets must use the same MAC address or they will be silently rejected by the fabric interconnect. In effect, this option enables port security for the associated vNIC.
If you plan to install VMware ESX on the associated server, you must configure the MAC Security to allow for the network control policy applied to the default vNIC. If you do not configure MAC Security for allow , the ESX installation may fail because the MAC security permits only one MAC address while the installation process requires more than one MAC address. |
| Step 8 | Click OK . |
-

Deleting a Network Control Policy

SUMMARY STEPS

1. In the **Navigation** pane, click **LAN**.
2. Expand **LAN > Policies > *Organization_Name***.
3. Expand the **Network Control Policies** node.
4. Right-click the policy you want to delete and select **Delete**.
5. If a confirmation dialog box displays, click **Yes**.

DETAILED STEPS

-
- | | |
|---------------|--|
| Step 1 | In the Navigation pane, click LAN . |
| Step 2 | Expand LAN > Policies > <i>Organization_Name</i> . |
| Step 3 | Expand the Network Control Policies node. |
| Step 4 | Right-click the policy you want to delete and select Delete . |
| Step 5 | If a confirmation dialog box displays, click Yes . |
-

Configuring Multicast Policies

Multicast Policy

This policy is used to configure Internet Group Management Protocol (IGMP) snooping and IGMP querier. IGMP Snooping dynamically determines hosts in a VLAN that should be included in particular multicast transmissions. You can create, modify, and delete a multicast policy that can be associated to one or more VLANs. When a multicast policy is modified, all VLANs associated with that multicast policy are re-processed to apply the changes. For private VLANs, you can set a multicast policy for primary VLANs but not for their associated isolated VLANs due to a Cisco NX-OS forwarding implementation.

By default, IGMP snooping is enabled and IGMP querier is disabled. When IGMP snooping is enabled, the fabric interconnects send the IGMP queries only to the hosts. They do not send IGMP queries to the upstream network. To send IGMP queries to the upstream, do one of the following:

- Configure IGMP querier on the upstream fabric interconnect with IGMP snooping enabled
- Disable IGMP snooping on the upstream fabric interconnect
- Change the fabric interconnects to switch mode

The following limitations and guidelines apply to multicast policies:

- On a 6200 series fabric interconnect, user-defined multicast policies can also be assigned along with the default multicast policy.
- Only the default multicast policy is allowed for a global VLAN.

- If a Cisco UCS domain includes 6300 and 6200 series fabric interconnects, any multicast policy can be assigned.
- We highly recommend you use the same IGMP snooping state on the fabric interconnects and the associated LAN switches. For example, if IGMP snooping is disabled on the fabric interconnects, it should be disabled on any associated LAN switches as well.

Creating a Multicast Policy

SUMMARY STEPS

1. In the **Navigation** pane, click **LAN**.
2. Expand **LAN > Policies**.
3. Expand the **root** node.
4. Right-click the **Multicast Policies** node and select **Create Multicast Policy**.
5. In the **Create Multicast Policy** dialog box, specify the name and IGMP snooping information.
6. Click **OK**.

DETAILED STEPS

-
- Step 1** In the **Navigation** pane, click **LAN**.
- Step 2** Expand **LAN > Policies**.
- Step 3** Expand the **root** node.
- Step 4** Right-click the **Multicast Policies** node and select **Create Multicast Policy**.
- Step 5** In the **Create Multicast Policy** dialog box, specify the name and IGMP snooping information.

- Note** Follow these guidelines if you choose to set IGMP Snooping querier IP addresses for a multicast policy:
- a. In the Ethernet Switch-Mode configuration, you must set the querier IP addresses for each FI in the domain.
 - b. In the Ethernet End-Host mode, you can set the querier IP address just for FI A, and optionally for FI B as well. If an IP address is not set explicitly for FI-B, it uses the same address set for FI A.
- Querier IP address can be any valid IP address. However, IP address from same subnet is required if there is a strict subnet check in the host.

- Step 6** Click **OK**.
-

Modifying a Multicast Policy

This procedure describes how to change the IGMP snooping state and the IGMP snooping querier state of an existing multicast policy.



-
- Note** You cannot change the name of the multicast policy once it has been created.
-

SUMMARY STEPS

1. In the **Navigation** pane, click **LAN**.
2. Expand **LAN > Policies**.
3. Expand the **root** node.
4. Click the policy that you want to modify.
5. In the work pane, edit the fields as needed.
6. Click **Save Changes**.

DETAILED STEPS

-
- | | |
|---------------|---|
| Step 1 | In the Navigation pane, click LAN . |
| Step 2 | Expand LAN > Policies . |
| Step 3 | Expand the root node. |
| Step 4 | Click the policy that you want to modify. |
| Step 5 | In the work pane, edit the fields as needed. |
| Step 6 | Click Save Changes . |
-

Deleting a Multicast Policy

**Note**

If you assigned a non-default (user-defined) multicast policy to a VLAN and then delete that multicast policy, the associated VLAN inherits the multicast policy settings from the default multicast policy until the deleted policy is re-created.

SUMMARY STEPS

1. In the **Navigation** pane, click **LAN**.
2. Expand **LAN > Policies**.
3. Expand the **root** node.
4. Right-click the **Multicast Policies** node and select **Delete Multicast Policy**.
5. If a confirmation dialog box displays, click **Yes**.

DETAILED STEPS

-
- | | |
|---------------|--|
| Step 1 | In the Navigation pane, click LAN . |
| Step 2 | Expand LAN > Policies . |
| Step 3 | Expand the root node. |
| Step 4 | Right-click the Multicast Policies node and select Delete Multicast Policy . |
| Step 5 | If a confirmation dialog box displays, click Yes . |
-

Configuring LACP Policies

LACP Policy

Link Aggregation combines multiple network connections in parallel to increase throughput and to provide redundancy. Link aggregation control protocol (LACP) provides additional benefits for these link aggregation groups. Cisco UCS Manager enables you to configure LACP properties using LACP policy.

You can configure the following for a lacp policy:

- **Suspended-individual:** If you do not configure the ports on an upstream switch for lacp, the fabric interconnects treat all ports as uplink Ethernet ports to forward packets. You can place the lacp port in suspended state to avoid loops. When you set suspend-individual on a port-channel with LACP, if a port that is part of the port-channel does not receive PDUs from the peer port, it will go into suspended state.
- **Timer values:** You can configure rate-fast or rate-normal. In rate-fast configuration, the port is expected to receive 1 PDU every 1 second from the peer port. The time out for this is 3 seconds. In rate-normal configuration, the port is expected to receive 1 PDU every 30 seconds. The timeout for this is 90 seconds.

System creates a default LACP policy at system start up. You can modify this policy or create a new policy. You can also apply one LACP policy to multiple port-channels.

Creating a LACP Policy

-
- | | |
|---------------|--|
| Step 1 | In the Navigation pane, click LAN . |
| Step 2 | Expand LAN > Policies . |
| Step 3 | Expand the node for the organization where you want to create the policy.

If the system does not include multitenancy, expand the root node. |
| Step 4 | In the Work Pane , click LACP Policies tab, and click the + sign. |
| Step 5 | In the Create LACP Policy dialog box, fill in the required fields. |
| Step 6 | Click OK . |
-

Modifying a LACP Policy

-
- | | |
|---------------|--|
| Step 1 | In the Navigation pane, click LAN . |
| Step 2 | Expand LAN > Policies . |
| Step 3 | Expand the node for the organization where you want to create the policy.

If the system does not include multitenancy, expand the root node. |
| Step 4 | In the Work Pane , LACP Policies tab, and click on the policy you want to edit. |
| Step 5 | Click the Properties icon on the right. |

Step 6 In the **Properties** dialog box, make the required changes and click **Apply**.

Step 7 Click **OK**.

Configuring UDLD Link Policies

Understanding UDLD

UniDirectional Link Detection (UDLD) is a Layer 2 protocol that enables devices connected through fiber-optic or twisted-pair Ethernet cables to monitor the physical configuration of the cables and detect when a unidirectional link exists. All connected devices must support UDLD for the protocol to successfully identify and disable unidirectional links. When UDLD detects a unidirectional link, it marks the link as unidirectional. Unidirectional links can cause a variety of problems, including spanning-tree topology loops.

UDLD works with the Layer 1 mechanisms to determine the physical status of a link. At Layer 1, autonegotiation takes care of physical signaling and fault detection. UDLD performs tasks that autonegotiation cannot perform, such as detecting the identities of neighbors and shutting down misconnected interfaces. When you enable both autonegotiation and UDLD, the Layer 1 and Layer 2 detections work together to prevent physical and logical unidirectional connections and the malfunctioning of other protocols.

A unidirectional link occurs whenever traffic sent by a local device is received by its neighbor but traffic from the neighbor is not received by the local device.

Modes of Operation

UDLD supports two modes of operation: normal (the default) and aggressive. In normal mode, UDLD can detect unidirectional links due to misconnected interfaces on fiber-optic connections. In aggressive mode, UDLD can also detect unidirectional links due to one-way traffic on fiber-optic and twisted-pair links and to misconnected interfaces on fiber-optic links.

In normal mode, UDLD detects a unidirectional link when fiber strands in a fiber-optic interface are misconnected and the Layer 1 mechanisms do not detect this misconnection. If the interfaces are connected correctly but the traffic is one way, UDLD does not detect the unidirectional link because the Layer 1 mechanism, which is supposed to detect this condition, does not do so. In case, the logical link is considered undetermined, and UDLD does not disable the interface. When UDLD is in normal mode, if one of the fiber strands in a pair is disconnected and autonegotiation is active, the link does not stay up because the Layer 1 mechanisms did not detect a physical problem with the link. In this case, UDLD does not take any action, and the logical link is considered undetermined.

UDLD aggressive mode is disabled by default. Configure UDLD aggressive mode only on point-to-point links between network devices that support UDLD aggressive mode. With UDLD aggressive mode enabled, when a port on a bidirectional link that has a UDLD neighbor relationship established stops receiving UDLD packets, UDLD tries to reestablish the connection with the neighbor and administratively shuts down the affected port. UDLD in aggressive mode can also detect a unidirectional link on a point-to-point link on which no failure between the two devices is allowed. It can also detect a unidirectional link when one of the following problems exists:

- On fiber-optic or twisted-pair links, one of the interfaces cannot send or receive traffic.
- On fiber-optic or twisted-pair links, one of the interfaces is down while the other is up.

- One of the fiber strands in the cable is disconnected.

Methods to Detect Unidirectional Links

UDLD operates by using two mechanisms:

- Neighbor database maintenance

UDLD learns about other UDLD-capable neighbors by periodically sending a hello packet (also called an advertisement or probe) on every active interface to keep each device informed about its neighbors. When the switch receives a hello message, it caches the information until the age time (hold time or time-to-live) expires. If the switch receives a new hello message before an older cache entry ages, the switch replaces the older entry with the new one.

UDLD clears all existing cache entries for the interfaces affected by the configuration change whenever an interface is disabled and UDLD is running, whenever UDLD is disabled on an interface, or whenever the switch is reset. UDLD sends at least one message to inform the neighbors to flush the part of their caches affected by the status change. The message is intended to keep the caches synchronized.

- Event-driven detection and echoing

UDLD relies on echoing as its detection mechanism. Whenever a UDLD device learns about a new neighbor or receives a resynchronization request from an out-of-sync neighbor, it restarts the detection window on its side of the connection and sends echo messages in reply. Because this behavior is the same on all UDLD neighbors, the sender of the echoes expects to receive an echo in reply.

If the detection window ends and no valid reply message is received, the link might shut down, depending on the UDLD mode. When UDLD is in normal mode, the link might be considered undetermined and might not be shut down. When UDLD is in aggressive mode, the link is considered unidirectional, and the interface is shut down.

If UDLD in normal mode is in the advertisement or in the detection phase and all the neighbor cache entries are aged out, UDLD restarts the link-up sequence to resynchronize with any potentially out-of-sync neighbors.

If you enable aggressive mode when all the neighbors of a port have aged out either in the advertisement or in the detection phase, UDLD restarts the link-up sequence to resynchronize with any potentially out-of-sync neighbor. UDLD shuts down the port if, after the fast train of messages, the link state is still undetermined.

UDLD Configuration Guidelines

The following guidelines and recommendations apply when you configure UDLD:

- A UDLD-capable interface also cannot detect a unidirectional link if it is connected to a UDLD-incapable port of another switch.
- When configuring the mode (normal or aggressive), make sure that the same mode is configured on both sides of the link.
- UDLD should be enabled only on interfaces that are connected to UDLD capable devices. The following interface types are supported:
 - Ethernet uplink
 - FCoE uplink
 - Ethernet uplink port channel member

- FCoE uplink port channel member

Creating a Link Profile

- Step 1** In the **Navigation** pane, click **LAN**.
- Step 2** Expand **LAN > Policies > LAN Cloud**.
- Step 3** Right-click the **Link Profile** node and choose **Create Link Profile**.
- Step 4** In the **Create Link Profile** dialog box, specify the name and the UDLD link policy.
- Step 5** Click **OK**.
-

Creating a UDLD Link Policy

- Step 1** In the **Navigation** pane, click **LAN**.
- Step 2** Expand **LAN > Policies > LAN Cloud**.
- Step 3** Right-click the **UDLD Link Policy** node and choose **Create UDLD Link Policy**.
- Step 4** In the **Create UDLD Link Policy** dialog box, specify the name, admin state, and mode.
- Step 5** Click **OK**.
-

Modifying the UDLD System Settings

- Step 1** In the **Navigation** pane, click **LAN**.
- Step 2** Expand **LAN > Policies > LAN Cloud**.
- Step 3** On the **LAN** tab, expand **LAN > Policies > root**.
- Step 4** Expand the **Link Protocol Policy** node and click **UDLD System Settings**.
- Step 5** In the **Work** pane, click the **General** tab.
- Step 6** In the **Properties** area, modify the fields as needed.
- Step 7** Click **Save Changes**.
-

Assigning a Link Profile to a Port Channel Ethernet Interface

- Step 1** In the **Navigation** pane, click **LAN**.
- Step 2** Expand **LAN > LAN Cloud > Fabric > Port Channels**.
- Step 3** Expand the port channel node and click the Eth Interface where you want to assign a link profile.

- Step 4** In the **Work** pane, click the **General** tab.
- Step 5** In the **Properties** area, choose the link profile that you want to assign.
- Step 6** Click **Save Changes**.
-

Assigning a Link Profile to an Uplink Ethernet Interface

- Step 1** In the **Navigation** pane, click **LAN**.
- Step 2** On the **LAN** tab, expand **LAN > LAN Cloud > Fabric > Uplink Eth Interface**.
- Step 3** Click the Eth Interface where you want to assign a link profile.
- Step 4** In the **Work** pane, click the **General** tab.
- Step 5** In the **Properties** area, choose the link profile that you want to assign.
- Step 6** Click **Save Changes**.
-

Assigning a Link Profile to a Port Channel FCoE Interface

- Step 1** In the **Navigation** pane, click **SAN**.
- Step 2** On the **SAN** tab, expand **SAN > SAN Cloud > Fabric > FCoE Port Channels**.
- Step 3** Expand the FCoE port channel node and click the FCoE Interface where you want to assign a link profile.
- Step 4** In the **Work** pane, click the **General** tab.
- Step 5** In the **Properties** area, choose the link profile that you want to assign.
- Step 6** Click **Save Changes**.
-

Assigning a Link Profile to an Uplink FCoE Interface

- Step 1** In the **Navigation** pane, click **SAN**.
- Step 2** On the **SAN** tab, expand **SAN > SAN Cloud > Fabric > Uplink FC Interfaces**.
- Step 3** Click the FCoE interface where you want to assign a link profile.
- Step 4** In the **Work** pane, click the **General** tab.
- Step 5** In the **Properties** area, choose the link profile that you want to assign.
- Step 6** Click **Save Changes**.
-

Configuring VMQ and VMMQ Connection Policies

VMQ Connection Policy

Cisco UCS Manager enables you to configure VMQ connection policy for a vNIC. VMQ provides improved network performance to the entire management operating system. Configuring a VMQ vNIC connection policy involves the following:

- Create a VMQ connection policy
- Create a static vNIC in a service profile
- Apply the VMQ connection policy to the vNIC

If you want to configure the VMQ vNIC on a service profile for a server, at least one adapter in the server must support VMQ. Make sure the servers have at least one the following adapters installed:

- UCS-VIC-12XX
- UCS-VIC-13XX
- UCS-VIC-14XX

The following are the supported Operating Systems for VMQ:

- Windows 2012
- Windows 2012R2
- Windows 2016

**Note**

The UCS-VIC-14XX adapter is not supported on Windows 2012 VMQ and Windows 2012 R2 VMQ

You can apply only any one of the vNIC connection policies on a service profile at any one time. Make sure to select one of the three options such as Dynamic, usNIC or VMQ connection policy for the vNIC. When a VMQ vNIC is configured on service profile, make sure you have the following settings:

- Select SRIOV in the BIOS policy.
- Select Windows in the Adapter policy.

Creating a VMQ Connection Policy

Before you create a VMQ connection policy, consider the following:

- VMQ Tuning on the Windows Server—When an adapter is placed on a virtual switch, running the **Get-NetAdapterVmq** cmdlet displays **True** for VMQ.
- Virtual machine level—By default, VMQ is enabled on all newly deployed VMs. VMQ can be enabled or disabled on existing VMs.

- Microsoft SCVMM — VMQ must be enabled on the port profile. If not, you will not be able to successfully create the virtual switch in SCVMM.
- Microsoft Azure Stack extends the existing VMQ support for host-side virtual switch ports called vPorts to Virtual Machine Multi Queues (VMMQ). You can configure VMMQ by enabling multi queues in the VMQ Connection Policy.

For the VIC 14XX adapter to support VMQ functionality, the vNIC should be configured in the VMQ connection policy with the multi-queue option enabled.



Note Microsoft Stand-alone NIC Teaming and Virtual Machine Queue (VMQ) support for VIC14xx adapters: Microsoft stand-alone NIC teaming works only with VMQ. For VIC 14xx adapters, the supported VMQ is VMMQ with single queue. To support VMMQ with single queue, you must create a new VMMQ adapter policy containing a 1 TQ, 1 RQ and 2 CQ combination, then assign it to the VMQ Connection Policy.

- Step 1** In the **Navigation** pane, click **LAN**.
- Step 2** In the **LAN** tab, expand **Policies**.
- Step 3** Expand the nodes for the organization where you want to create the policy. If the system does not include multitenancy, expand the root node.
- Step 4** Right-click the **VMQ Connection Policies** node and select **Create VMQ Connection Policy**.
- Step 5** In the **Create VMQ Connection Policy** dialog box, complete the following fields:

Name	Description
Name field	The VMQ connection policy name.
Description field	The description of the VMQ connection policy.

Name	Description
Multi Queue radio button	<p>Whether Virtual Machine Multi-Queue (VMMQ) is enabled in the policy. With VMMQ, multiple queues are allocated to a single VM.</p> <ul style="list-style-type: none"> • Disabled—Multi Queue is disabled and you can configure a VMQ policy. <p>When Multi Queue is disabled, the following fields appear:</p> <ul style="list-style-type: none"> • Number of VMQs • Number of Interrupts <ul style="list-style-type: none"> • Enabled—Multi Queue is enabled and the vNIC is placed into VMMQ mode. You can specify a VMMQ Adapter Policy. <p>When Multi Queue is enabled, the following fields appear:</p> <ul style="list-style-type: none"> • Number of Sub vNICs • VMMQ Adapter Policy <p>Note For VIC 14XX adapters, enable the Multi-Queue option to support both VMQ/VMMQ functionality.</p> <p>For more information on creating a VMQ Connection Policy with Multi-Queue enabled, please see Creating a VMMQ Connection Policy, on page 223.</p>
Number of VMQs field	<p>The number of VMQs per adapter must be one more than the maximum number of VM NICs. The default value is 64.</p> <p>Note Make sure that the total number of synthetic NICs present on the VMs is either equal to or greater than the number of VMs.</p>
Number of Interrupts field	<p>The number of CPU threads or logical processors available in the server. The default value is 64.</p> <p>Note You cannot set this value to be more than the maximum number of available CPUs.</p>

Step 6 Click **OK**.

Assigning VMQ Setting to a vNIC

SUMMARY STEPS

1. In the **Navigation** pane, click **Servers**.
2. On the **Servers** tab, expand **Servers > Service Pprofile > root**.
3. Expand the service profile that you want to configure for VMQ and then click **vNICs**.
4. In the **Work Pane**, click the **Network** tab.
5. In the **vNICs** area, choose a vNIC and double-click the **Actual Order** column.
6. In the **Adapter Performance Profile** area of the Modify vNIC dialog box, choose **Windows** from the Adapter Policy drop-down list.
7. In the **Connection Policies** area, click the **VMQ** radio button.
8. Select the **VMQ Connection Policy** from the VMQ Connection Policy drop-down list.
9. Click **OK**.
10. Click **Save Changes**.

DETAILED STEPS

-
- | | |
|----------------|---|
| Step 1 | In the Navigation pane, click Servers . |
| Step 2 | On the Servers tab, expand Servers > Service Pprofile > root . |
| Step 3 | Expand the service profile that you want to configure for VMQ and then click vNICs . |
| Step 4 | In the Work Pane , click the Network tab. |
| Step 5 | In the vNICs area, choose a vNIC and double-click the Actual Order column.
Modify vNIC window is displayed. |
| Step 6 | In the Adapter Performance Profile area of the Modify vNIC dialog box, choose Windows from the Adapter Policy drop-down list. |
| Step 7 | In the Connection Policies area, click the VMQ radio button. |
| Step 8 | Select the VMQ Connection Policy from the VMQ Connection Policy drop-down list. |
| Step 9 | Click OK . |
| Step 10 | Click Save Changes . |
-

Enabling VMQ and NVGRE Offloading on the same vNIC

Perform the tasks in the table below to enable VMQ and NVGRE offloading on the same vNIC.



Note VMQ is not supported along with VXLAN on the same vNIC except for VIC 14XX adapters. VIC 14XX supports VMQ/VMMQ along with VXLAN or NVGRE on the same vNIC.

Task	Description	See
Enable normal NVGRE offloading	Perform this task by setting the corresponding flags in the adapter profile which is associated with the given vNIC. Note The Transmit checksum offload and TSO must be enabled for the NVGRE offloading to be effective.	Configuring an Ethernet Adapter Policy to Enable Stateless Offloads with NVGRE, on page 195
Enable VMQ	Perform this task by setting the appropriate connection policy when you add a vNIC to the service profile.	Creating a VMQ Connection Policy, on page 218 Assigning VMQ Setting to a vNIC, on page 221

VMMQ Connection Policy

Cisco UCS Manager introduces support for Virtual Machine Multi Queues (VMMQ). VMMQ allows you to configure multiple I/O queues to a single VM and, thus, distribute traffic across multiple CPU cores in a VM. VMMQ is supported on UCS VIC 14xx adapters only with Windows 2016.

The VMQ Connection Policy has an option called **Multi Queue**. When **Multi Queue** is enabled, the vNIC is placed into VMMQ mode. In this mode, you can configure sub vNICs and specify a VMMQ Adapter policy. The policy includes the aggregate queue counts for VMMQ and determines how the connectivity between VMs and Azure Stack vPorts is configured.

There are two different ways to define the total number of queues available for vPorts. In the pooled mode, the resource counts in the VMMQ adapter policy are the totals available across vPorts. In non-pooled mode, the total available is the selected resource count from the VMMQ adapter policy * subvnic count. In VMMQ mode, these are the default queue counts:

Queue Resource	Pooled Mode	Non Pooled Mode
Transmit Queue	64	1
Receive Queue	512	8
Completion Queue	576	9

[Creating a VMMQ Connection Policy, on page 223](#) provides detailed information about creating a VMMQ connection policy.

VMMQ Guidelines

- Each VMMQ vPort may use multiple Transmit and Receive Queues. When VMMQ is enabled, a pool of queues is created, and the host driver assigns queues to vPorts. Different vPorts may be assigned different numbers of queues based on the number of cores that the vPort will be servicing.

- VXLAN and NVGRE offloads are supported with VMMQ functionality. The option is enabled in the vNIC adapter policy and not in the sub vNIC adapter policy.
- RSS is supported on VMMQ Receive Queues, including inner packet of overlay packets.
- VMMQ vNICs support a rate limit set by the host, not from Cisco UCS Manager. COS will not be adjustable per vPort from Cisco UCS Manager.
- vNICs with the VMQ feature, specified through the VMQ Connection Policy with **Multi Queue** disabled, are not allowed on the same adapter as Multi Queue-enabled vNICs.
- Netflow may be enabled on the vNIC that is enabled for VMMQ. The counts reported will be the aggregated counts across all vPorts. Netflow cannot distinguish between flows from one vPort and another.
- FCoE and VMMQ vNICs can coexist on the same server.
- usNIC and Multi-Queue VMQ can not be enabled on the same VIC.
- Modifying the VMMQ adapter policy through the VMQ connection policy results in exceeding the maximum Completion Queue (CQ) value. Each VIC 1400 Series adapter supports a maximum of 2000 hardware CQ resources. If this number is exceeded, the `Out of CQ Resources` error appears in the Cisco UCS Manager GUI, and vNIC creation fails with a configuration failure at service profile association.
- By default, only VMQ is enabled on all newly deployed VMs. To enable VMMQ support, the following PS command needs to be run on Host server.

```
Set-VMNetworkAdapter -Name (vmNIC Name) -VMName (VM_NAME) -VmmqEnabled $true
-VmmqQueuePairs (Queue_Pair_Count) -VrssEnabled $true
```

Creating a VMMQ Connection Policy

VMMQ connection policy can be created using VMQ policy with Multi Queue enabled.

- Step 1** In the **Navigation** pane, click **LAN**.
- Step 2** In the **LAN** tab, expand **Policies**.
- Step 3** Expand the nodes for the organization where you want to create the policy. If the system does not include multitenancy, expand the root node.
- Step 4** Right-click the **VMQ Connection Policies** node and select **Create VMQ Connection Policy**.
- Step 5** In the **Create VMQ Connection Policy** dialog box, complete the following fields:

Name	Description
Name field	The VMQ connection policy name.
Description field	The description of the VMQ connection policy.

Name	Description
Multi Queue radio button	<p>When Virtual Machine Multi-Queue (VMMQ) is enabled in the policy, multiple queues are allocated to a single VM.</p> <ul style="list-style-type: none"> • Enabled—Multi Queue is enabled and the vNIC is placed into VMMQ mode. You can specify a VMMQ Adapter Policy. <p>When Multi Queue is enabled, the following fields appear:</p> <ul style="list-style-type: none"> • Number of Sub vNICs • VMMQ Adapter Policy <p>Note For VIC 14XX adapters, enable the Multi-Queue option to support both VMQ/VMMQ functionality.</p>
Number of Sub vNICs field	<p>Number of sub vNICs that are available for Multi Queue. The default value is 64.</p> <p>Note The TQ and RQ resource value of VMMQ adapter policy should be greater than or equal to the configured number of sub vNICs.</p>
VMMQ Adapter Policy drop-down list	<p>Name of the VMMQ adapter policy. Cisco recommends using MQ Adapter Policy.</p> <p>The default MQ policy includes the aggregate queue counts for VMMQ.</p>

Step 6 Click **OK**.

Creating a QoS Policy for VMMQ

- Step 1** In the **Navigation** pane, click **LAN**.
- Step 2** In the **LAN** tab, expand **Policies**.
- Step 3** Expand the nodes for the organization where you want to create the pool. If the system does not include multitenancy, expand the **root** node.
- Step 4** Right-click the **QoS Policy** dialog box and enter the name of the policy in the **Name** field.
- Step 5** Select the desired priority in the **Priority** drop-down list.
- Step 6** In the **Host Control** field, click the **Full** radio button.
- Step 7** Click **OK**.

Assigning a VMMQ Setting to a vNIC

-
- Step 1** In the **Navigation** pane, click **Servers**.
- Step 2** In the **Servers** tab, expand **Servers > Service Profiles > root**.
- Step 3** Expand the service profile that you want to configure VMMQ and click **vNICs**.
- Step 4** In the **Work Pane**, click the **Network** tab.
- Step 5** In the **vNICs** area, choose the desired vNIC and double-click the **Actual Order** column.
Modify vNIC window is displayed.
- Step 6** In the **Adapter Performance Profile** area of the **Modify vNIC** dialog box, choose MQ from the **Adapter Policy** drop-down list.
- Step 7** From the **QoS Policy** drop-down list, select the created QoS policy for VMMQ.
- Step 8** In the **Connection Policies** area, click the **VMQ** radio button.
- Step 9** Choose the created VMQ connection policy with Multi-Queue enabled from the **VMQ Connection Policy** drop-down list.
- Step 10** Click **OK**.
- Step 11** Click **Save Changes**.
-

NetQueue

Information About NetQueue

NetQueue improves traffic performance by providing a network adapter with multiple receive queues. These queues allow the data interrupt processing that is associated with individual virtual machines to be grouped.



Note NetQueue is supported on servers running VMware ESXi operating systems.

Configuring NetQueue

-
- Step 1** In the **Navigation** pane, click **LAN**.
- Step 2** In the **LAN** tab, expand **Policies**.
- Step 3** Expand the nodes for the organization where you want to create the policy. If the system does not include multitenancy, expand the root node.
- Step 4** Right-click the **VMQ Connection Policies** node and select **Create VMQ Connection Policy**.
- Step 5** In the **Create VMQ Connection Policy** dialog box, complete the following fields:

Step 6

Name	Description
Name field	The NetQueue policy name.
Description field	The description of the NetQueue.
Multi Queue radio button	Select disabled for NetQueue.
Number of VMQs field	<p>Enter a number between 1 to 64 to specify the number of NetQueues for this connection policy. The driver supports up to 16 NetQueues per port for standard frame configurations.</p> <p>Note VMware recommends that you use up to eight NetQueues per port for standard frame configurations.</p>
Number of Interrupts field	The number of interrupts count of each VNIC. The value should be set to 2 x number of VMQs + 2.

Step 7 Click **OK**.

Step 8 In the **Navigation** pane, click **Servers**.

Step 9 On the **Servers** tab, expand **Servers > Service Profiles > root**.

Step 10 Expand the service profile that you want to configure NetQueue and click vNICs.

Step 11 In the **Work** pane, click the **Network** tab.

Step 12 In the vNICs area, choose a vNIC and double-click the **Actual Order** column.

Modify vNIC window is displayed.

Step 13 In the **Adapter Performance Profile** area of the Modify vNIC dialog box, choose **VMWare** from the Adapter Policy drop-down list.

Step 14 In the **Connection Policies** area, click the **VMQ** radio button.

Step 15 Choose the created VMQ connection policy for NetQueue from the VMQ Connection Policy drop-down list.

Step 16 Click **OK**.

Step 17 Click **Save Changes**.

Note NetQueue should be enabled only on MSIX systems.

You should disable NetQueue on 1GB NICs.